FINITE ELEMENT MODEL UPDATING THROUGH SMOOTH NONCONVEX OPTIMIZATION

A Dissertation
Presented to
The Academic Faculty

by

Xinjun Dong

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Civil and Environmental Engineering

Georgia Institute of Technology
May, 2019

FINITE ELEMENT MODEL UPDATING THROUGH SMOOTH NONCONVEX OPTIMIZATION

Approved by:

Dr. Yang Wang, Advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Francesco Fedele, Co-advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Rafi L. Muhanna, Co-advisor
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Glaucio Paulino
School of Civil and Environmental Engineering
Georgia Institute of Technology

Dr. Santanu S. Dey
School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Phanish Suryanarayana
School of Civil and Environmental Engineering
Georgia Institute of Technology

Date Approved: [December 27, 2018]

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my advisor, Dr. Yang Wang, for his continuous support and guidance throughout my Ph.D. life. It has been an honor to be his Ph.D. student. He has taught me how to grow as a researcher. I appreciate all his contributions of time, ideas, and funding to make my Ph.D. experience productive and stimulating. The joy and enthusiasm he has for his research was contagious and motivational for me.

I would also like to thank Dr. Francesco Fedele, Dr. Santanu S. Dey, Dr. Rafi L. Muhanna, Dr. Glaucio Paulino and Dr. Phanish Suryanarayana for sitting on my dissertation committee, and for their valuable time and support of my research.

I want to thank Dapeng Zhu for his guidance on finite element model updating. I would like to thank Dan Li and Xi Liu for their assistance on developing and validating the wireless sensing system. Sincere thanks are also given to my colleagues and friends at Georgia Tech, including Chunhee Cho, Xiaohua Yi, Yidong Qin, Ke Liu and all others that I have not listed here, for helping me with various aspects of my research and life.

Last but no means least, I thank my wife, Xia Li, for her love and support at all times. Special thanks also owe to my parents, Jianzhong Dong and Yan Ding, for their encouragement and unconditional trust and love.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS i

LIST OF TABLES iv

LIST OF FIGURES vi

LIST OF SYMBOLS AND ABBREVIATIONS x

SUMMARY xii

CHAPTER 1. Introduction 1

1.1 Introduction of FE Model Updating Approaches 1

1.2 Introduction of Optimization Algorithms 6

1.3 Organization of Dissertation 9

CHAPTER 2. FE model updating formulations and local search optimization algorithms 11

2.1 Convex Optimization 11

2.2 FE Model Updating Formulations 13
2.2.1 Modal Property Difference Formulation with MAC Values 15
2.2.2 Modal Property Difference Formulation with Eigenvector Difference 17
2.2.3 Modal Dynamic Residual Formulation 18

2.3 Jacobian Derivative of Model Updating Formulations 19
2.3.1 Jacobian of MAC Value Formulation 20
2.3.2 Jacobian of Eigenvector Difference Formulation 24
2.3.3 Jacobian of Modal Dynamic Residual Formulation 25

2.4 Local Search Optimization Algorithms 27

2.5 Numerical Studies 32
2.5.1 Lumped Mass-Spring Structure 33
2.5.2 Steel Pedestrian Bridge 42
2.5.3 Concrete Building Frame 50

CHAPTER 3. FE model updating with experimental data 61

3.1 *Martlet* Wireless Sensing Node 62
3.1.1 Martlet Node 62
3.1.2 Integrated Accelerometer Wing 64

3.2 FE Model Updating with Laboratory Experiment Data 67
3.2.1 Description of Test Structure and Test Configuration 67
3.2.2 Test Results and Modal Analysis 68
3.2.3 FE Model Updating Results 71

3.3 FE Model Updating with Field Experiment Data 80
3.3.1 Test Configuration and Test Results 81

3.3.2 Modal Analysis Results 84
3.3.3 FE Model Updating of the Pedestrian Bridge 91

CHAPTER 4. FE model updating with global optimization algorithms 103

4.1 Branch-and-Bound Algorithm 104

4.2 Primal-Relaxed Dual Algorithm 109
4.2.1 Introduction of P-RD Algorithm 110
4.2.2 Pseudo Code for the P-RD Algorithm 119

4.3 FE Model Updating Formulations for Global Optimization Algorithms 124
4.3.1 Eigenvector Difference Formulation 124
4.3.2 Modal Dynamic Residual Formulation 127

4.4 Validation Example 128
4.4.1 Plane Truss Structure 128
4.4.2 Four-Story Shear-Frame Structure 135

CHAPTER 5. Summary and future work 142

5.1 Summary and Conclusions 142

5.2 Future Work 144

REFERENCES 147

LIST OF TABLES

Table 2-1 Model updating algorithms for objective function in Section 2.2 33

Table 2-2 Structural properties of the lumped mass-spring structure 34

Table 2-3 Structural properties of the steel pedestrian bridge 44

Table 2-4 Structural properties of the concrete building frame 52

Table 3-1 Structural properties of the four-story shear-frame structure 68

Table 3-2 Comparison of experimental and simulated modal properties from the nominal model 72

Table 3-3 Structural properties of the shear-frame structure 73

Table 3-4 Comparison of experimental and simulated modal properties from MAC value formulation 75

Table 3-5 Comparison of experimental and simulated modal properties from eigenvector difference formulation 77

Table 3-6 Comparison of experimental and simulated modal properties from modal dynamic residual formulation 78

Table 3-7 Comparison of optimal stiffness values among different model updating formulations 79

Table 3-8 Structural properties of the pedestrian bridge 93

Table 3-9 Comparison of experimental and simulated modal properties from the nominal model 94

Table 3-10 Comparison of experimental and simulated modal properties from MAC value formulation 98

Table 3-11 Comparison of experimental and simulated modal properties from eigenvector difference value formulation 100

Table 3-12 Comparison of updated model from different model updating formulations 101

Table 4-1 Structural properties of the plane-truss structure 129

Table 4-2 Comparison of experimental and simulated modal properties from the nominal model 136

Table 4-3 Structural properties of the shear-frame structure 136

Table 4-4 Comparison of experimental and simulated modal properties from the eigenvector difference formulation 139

Table 4-5 Comparison of experimental and simulated modal properties from the modal dynamic residual formulation 141

LIST OF FIGURES

Figure 1-1. Illustration of a nonconvex function 7

Figure 2-1. Illustration of a convex function 12

Figure 2-2. Illustration of a convex set 13

Figure 2-3. Lumped mass-spring model and sensor instrumentation 34

Figure 2-4. Lumped mass-spring model: updating results of MAC value formulation using analytical gradient 38

Figure 2-5. Lumped mass-spring model: updating results of eigenvector difference formulation using analytical gradient 39

Figure 2-6. Lumped mass-spring model: updating results of modal dynamic residual formulation using analytical gradient 40

Figure 2-7. Steel pedestrian bridge model and sensor instrumentation 43

Figure 2-8. Steel pedestrian bridge model: updating results of MAC value formulation using analytical gradient 46

Figure 2-9. Steel pedestrian bridge model: updating results of eigenvector difference formulation using analytical gradient 48

Figure 2-10. Steel pedestrian bridge model: updating results of modal dynamic residual formulation using analytical gradient 49

Figure 2-11. Model of a 2-story 2-bay concrete building frame [59] (height in z: 2 × 12 ft.; length in x: 2 ×18 ft.; width in y: 9 ft.) and sensor instrumentation 51

Figure 2-12. Concrete building frame model: updating results of MAC value formulation using numerical gradient 55

Figure 2-13. Concrete building frame model: updating results of MAC value formulation using analytical gradient 55

Figure 2-14. Concrete building frame model: updating results of eigenvector difference formulation using numerical gradient 56

Figure 2-15. Concrete building frame model: updating results of eigenvector difference formulation using analytical gradient 57

Figure 2-16. Concrete building frame model: updating results of modal dynamic residual formulation using analytical gradient 59

Figure 3-1. *Martlet* wireless sensor node (2.5 in × 2.25 in) 63

Figure 3-2. Integrated accelerometer *wing* with weatherproof package 64

Figure 3-3. Functional diagram of the intergrated accelerometer *wing* 65

Figure 3-4. Interface *wing* between *Martlet* node and integrated accelerometer *wing* (2.5 in × 2.25 in) 66

Figure 3-5. Experiment setup of the four-story shear-frame structure 67

Figure 3-6. Comparison between integrated *Martlet* accelerometer *wing* and cabled accelerometer 69

Figure 3-7. Example response records and corresponding frequency spectra records 70

Figure 3-8. Modal properties of first four modes of the shear-frame structure 71

Figure 3-9. Simplified model of four-story shear-frame structure 72

Figure 3-10. Shear frame structure: updating results of MAC value formulation using analytical gradient 75

Figure 3-11. Shear frame structure: updating results of eigenvector difference formulation using analytical gradient 76

Figure 3-12. Shear frame structure: updating results of modal dynamic residual formulation using analytical gradient 78

Figure 3-13. Pedestrain bridge on Georgia Tech campus 80

Figure 3-14. Experimental setup for the wirless sensor testing 81

Figure 3-15. Example vibration records and corresponding frequency spectra when hammer impact is applied laterally at H1 83

Figure 3-16. Example vibration records and corresponding frequency spectra when hammer impact is applied vertically at H1 84

Figure 3-17. Illustration of approximating a complex-valued vector with a real-valued vector 87

Figure 3-18. First four resultant modal proeprties of the pedestrain bridge 91

Figure 3-19. FE model of the pedestrian bridge 92

Figure 3-20. Pedestrian bridge model: updating results of MAC value formulation using analytical gradient 97

Figure 3-21. Pedestrian bridge model: updating results of eigenvector difference formulation using analytical gradient 99

Figure 4-1. Illustration of convex envelope 105

Figure 4-2. Projection of Prob. ① onto the space of variable y 112

Figure 4-3. Illustration of solving Prob. ④ 113

Figure 4-4. Illustration of P-RD algorithm (Initial condition) 114

Figure 4-5. Illustration of P-RD algorithm (1^{st} iteration) 117

Figure 4-6. Illustration of P-RD algorithm (2^{nd} iteration) 119

Figure 4-7. Pesudo code of constructing $\mathbf{R}_c$ 120

Figure 4-8. Illustration of P-RD algorithm (3^{rd} iteration) 120

Figure 4-9. Pseudo code of P-RD algorithm 124

Figure 4-10. Plane truss structure and sensor instrumentation 129

Figure 4-11. Plane truss structure: updating results of eigenvector difference formulation (local optimization algorithm) 131

Figure 4-12. Plane truss structure: updating results of eigenvector difference formulation (global optimization algorithms) 132

Figure 4-13. Plane truss structure: updating results of modal dynamic residual formulation (local optimization algorithm) 133

Figure 4-14. Plane truss structure: updating results of modal dynamic residual formulation (global optimization algorithms) 134

Figure 4-15. Modal properties of first two modes of the shear-frame structure 135

Figure 4-16. Shear frame structure: updating results of eigenvector difference formulation (local optimization algorithm) 138

Figure 4-17. Shear frame structure: updating results of eigenvector difference formulation (global optimization algorithms) 138

Figure 4-18. Shear frame structure: updating results of modal dynamic residual formulation (local optimization algorithm) 140

Figure 4-19. Shear frame structure: updating results of modal dynamic residual formulation (global optimization algorithms)..... 140

LIST OF SYMBOLS AND ABBREVIATIONS

$\mathbf{K}$	Stiffness matrix
$\boldsymbol{\alpha}$	Updating variables for stiffness parameters
$\mathbf{K}_0$	Nominal stiffness matrix
$\mathbf{K}_j$	Constant influence matrix
$\mathbf{M}$	Mass matrix
ω_i^{EXP}	*i*-th experimental resonance frequency
λ_i^{EXP}	*i*-th experimental eigenvalue
DOF	Degree of freedom
$\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$	*i*-th experimental mode shape at measured DOFs
λ_i	*i*-th simulated eigenvalue
$\boldsymbol{\psi}_i$	*i*-th simulated eigenvector
$\boldsymbol{\psi}_i^{\mathrm{m}}$	*i*-th simulated eigenvector at measured DOFs
$\boldsymbol{\psi}_i^{\mathrm{u}}$	*i*-th simulated eigenvector at unmeasured DOFs
n_{modes}	Number of experimentally measured modes
n_{m}	Number of measured DOFs
n_{u}	Number of unmeasured DOFs
N	Total number of DOFs
MAC	Modal assurance criterion
w_{λ_i}	Weighting factor of the eigenvalue difference
$w_{\boldsymbol{\psi}_i}$	Weighting factor of the eigenvector difference
$\mathbf{Q}_i$	Selection matrix
w_i	Weighting factor for the *i*-th modal residuals

B&B	Branch-and-Bound
P-RD	Primal-Relaxed Dual
f_u	Upper bound of optimal objective function value
f_l	Lower bound of optimal objective function value

SUMMARY

During the past few decades, great efforts have been devoted towards finite element (FE) modeling of structures, in order to simulate the structural behavior under various loading conditions. Due to the complexity of large-scale civil structures, the simulation results generated by an FE model are usually different from these of the as-built structure. To reduce the difference, selected structural parameters can be updated using the data collected from the actual structure. This process is known as FE model updating. An updated FE model can more accurately predict structural behavior under different loading conditions. In addition, FE model updating may be used to identify parameter value changes caused by structural deterioration or damage.

This research explores FE model updating utilizing the measured frequency-domain modal properties, i.e. resonance frequencies and mode shapes. Naturally, FE model updating is formulated as a mathematical optimization problem aiming to minimize the difference between simulated and experimentally-measured modal properties. In the first stage of this research, two categories of frequency-domain model updating approaches are studied, i.e. modal property difference and modal dynamic residual approaches. For each model updating approach, Jacobian derivative of the objective function is derived in detail. To find the optimal solution of the formulated optimization problem, two conventional local search algorithms are studied for comparison, namely the Levenberg-Marquardt and the trust-region reflective algorithms. The performance of the model updating approaches and optimization algorithms are first validated through numerical simulations. To further validate the presented model updating approaches, both laboratory and field experiments

are conducted. A new wireless sensing node, named *Martlet*, is installed on the structures to collect the vibration data. Experimental modal properties are extracted from the vibration data, and then used to perform the model updating.

Finally, in order to overcome the limitation of local search algorithms, this research also investigates global optimization algorithms to solve the optimization problems in FE model updating. Two global optimization algorithms, i.e. branch-and-bound (B&B) and primal-relaxed dual (P-RD) algorithms, are studied. Both B&B and P-RD algorithms attempt to approximate the lower and upper bounds of the optimal objective function value, and to reduce the gap between the lower and upper bounds in an iterative manner. Unlike the local search algorithms, the global optimization algorithms can guarantee the global optimality of the final solution. Again, the performance of the B&B and P-RD algorithms are validated through both numerical simulation and laboratory experiment.

CHAPTER 1. INTRODUCTION

Finite element (FE) modeling is an important tool for many engineering applications. However, despite significant progress made in FE modeling techniques, discrepancies inevitably exist between the behavior of a numerical model and of the as-built structure. The discrepancy may be caused by various approximation and inaccuracies in the models. For example, idealized connections and support conditions are typically used in structural analysis and design, while these conditions do not exist in reality. In addition, material properties of the actual structures are always different from the nominal values, particularly for concrete structures. To obtain a more accurate FE model that truly reflects behaviors of an actual constructed structure, data collected from the actual structure can be used to update the values of selected model parameters (e.g. stiffness or mass parameters). This process is known as FE model updating. An updated FE model can more accurately predict structural behavior under different loading conditions. In the meantime, benefiting from the development of low-cost wireless sensing systems [1-3], more and more structural sensors are becoming available for measuring structural responses. As a result, large amount of sensor data collected from actual structures are becoming available for FE model updating.

1.1 Introduction of FE Model Updating Approaches

Numerous FE model updating algorithms have been developed and practically applied in the past few decades [4]. Most algorithms can be categorized into two groups, i.e. time-domain and frequency-domain approaches. Time-domain approaches directly deal with time history data collected from the actual structure. For example, early

researchers started with extended Kalman filter (EKF) [5-7]. The EKF approach forms a state vector containing the displacement and velocity of all degrees of freedom (DOFs), as well as the selected structural parameters to be updated. The state vector is estimated through dynamic state propagation and corrected by minimizing the covariance of the estimation error of the formulated state vector. In addition, some time-domain approaches are capable of updating nonlinear structural hysteresis parameters. For example, the EKF approach is adopted to estimate the time-invariant parameters of a modified Giuffré-Menegotto-Pinto material constitutive model [8]. Furthermore, forgetting factors are adopted to identify the time-varying parameters of a nonlinear FE model [9, 10]. Overall, the time-domain approaches suffer convergence difficulties and high computational cost when applied to large-scale FE models.

Different from the time-domain approaches, frequency-domain approaches can update an FE model using frequency-domain modal properties extracted from experimental measurements, such as resonance frequencies, vibration mode shapes and damping ratios. The frequency-domain properties are described by the generalized eigenvalue equation in structural dynamics:

$$[\mathbf{K} - \omega_i^2 \mathbf{M}]\boldsymbol{\psi}_i = \mathbf{0} \tag{1-1}$$

where ω_i and $\boldsymbol{\psi}_i$ are the *i*-th resonance frequency and mode shape of the structure, respectively; $\mathbf{K}$ and $\mathbf{M}$ are the stiffness and mass matrices of the structure, which contain values of structural parameters to be updated. In particular, early researchers started by minimizing an objective function consisting of the differences between measured and simulated resonance frequencies (ω_i). This category of model updating approaches is

named as modal property difference approach. For example, Zhang *et al.* proposed an eigenvalue sensitivity-based model updating approach that was applied on a scaled suspension bridge model, and the updated FE model shows resonance frequencies closer to the experimental measurements [11]. Salawu reviewed various model updating formulations using natural frequency difference, and concluded that differences in frequencies may not be sufficient enough for accurately identifying structural parameter values [12]. Therefore, other modal properties, e.g. mode shapes or modal flexibility, were investigated for model updating. For example, Moller and Friberg adopted the modal assurance criterion (MAC)-related function for updating the model of an industrial structure, in attempt to make the updated FE model generate mode shapes that are closer to those extracted from experimental measurements [13]. Ribeiro *et al.* adopts the MAC value formulation to update the FE model of a railway bridge, and the numerical simulation demonstrates better agreement with experimental data after model updating [14]. FE model updating using differences in simulated and experimental mode shapes and frequencies was also applied to damage assessment of a reinforced concrete beam [15] and a reinforced concrete frame [16]. For damage assessment, the updated FE model for the undamaged structure is first taken as "healthy" baseline. If later model updating of the structure shows stiffness reduction of some structural components, the reduction can potentially be caused by damage (e.g. corrosion-induced section loss). Jaishi and Ren proposed a minimization objective function consisting of difference in modal flexibility for updating the model of a beam structure [17]. In addition, Zhou and Song adopts modal flexibility to update a pedestrian bridge model while considering the temperature effect on model updating process [18]. Aiming at practical applications, Yuen developed an efficient model updating

algorithm using frequencies and mode shapes at only a few selected degrees-of-freedom (DOFs) for the first few modes [19]. Naturally, the modal property difference approach is formulated as a mathematical optimization problem by setting the selected structural parameters as optimization variables.

Another main category in frequency-domain FE model updating is the modal dynamic residual approach [20]. Also formulated as a mathematical optimization problem, the modal dynamic residual approach aims to minimize the residuals of the generalized eigenvalue equation in structural dynamics (Eq. (1-1)). When calculating the residuals, experimentally-measured ω_i and $\boldsymbol{\psi}_i^{\mathrm{m}}$ are used in combination with the structural matrices, $\mathbf{K}$ and $\mathbf{M}$ assembled with the structural parameter values. We use $\boldsymbol{\psi}_i^{\mathrm{m}}$ to denote the entries corresponding to the measured DOFs in $\boldsymbol{\psi}_i$. Sanayei *et al*. formed an objective function consisting of modal dynamic residuals condensed to the measured DOFs, $[\mathbf{K}^{\mathrm{m}} - \omega_i^2\mathbf{M}^{\mathrm{m}}]\boldsymbol{\psi}_i^{\mathrm{m}}$ [21, 22]. Here $\mathbf{K}^{\mathrm{m}}$ and $\mathbf{M}^{\mathrm{m}}$ are the stiffness and mass matrix condensed to the measured DOFs, and Abdalla *et al*. formulated a linear matrix inequality (LMI) optimization problem that minimizes the change in stiffness matrix (from initial estimate) under constraints on the magnitude of modal dynamic residuals [23]. Recently, Zhu *et al.* applied the modal dynamic residual approach on substructure model updating, and were able to identify the structural parameters of a 3D frame structure [24].

A third main category in frequency-domain FE model updating is to formulate an optimization problem to minimize the difference between simulated and experimental frequency response functions (FRF) by setting stiffness, mass and damping parameters as optimization variables [25]. To facilitate the FRF model updating process, an analytical

sensitivity-based FRF updating approach is proposed by Lin and Ewins, by which a 2D FE model is efficiently updated [26]. Later, Lus *et al*. proposed a two-step methodology to update the physical parameters of the structure [27]. In the first step, a first-order state space realization of the system is updated using the input-output measurements, and modal properties of the system are extracted. The modal properties are then adopted to update the mass, stiffness and damping matrices of the linear dynamical system. Nevertheless, the FRF difference approach requires to record both excitation and structural response during the experiment, which is not applicable for ambient vibration data.

In practical applications, experimental data is inevitably contaminated with measurement noises. The noisy data causes uncertainties in model updating results, which negatively influence FE model updating accuracy. Researchers have investigated noise effect reduction for some FE model updating approaches. For example, Ahmadian *et al*. added the regularization of structural parameter changes in the objective function for FE model updating, which aims to balance the confidence in the model updating parameters and in the measurement data [28]. In addition, some studies consider the statistical properties of measurement noises and structural parameters in the model updating process. Bayesian statistical framework is widely adopted in FE model updating to reduce the influence of measurement noises by maximizing the likelihood of the experimental data or the *a posteriori* probability of structural parameters [29-31]. Vanik *et al*. applied the Bayesian framework on the modal property difference approach and identify the damage of a simulated shear-frame structure. In addition, using the Gibbs sampler method, Ching *et al*. solved the problem of Bayesian modal property difference approach and managed to update the FE model of an IASC-ASCE benchmark structure [32]. Later, Behmanesh and

Moaveni applied the Markov Chain Monte Carlo on the Bayesian modal property difference approach to update the FE model of a footbridge, and identify the simulated structural parameter changes [33]. Similar to the modal property difference and modal dynamic residual approaches, both the regularized and Bayesian model updating approaches can also be formulated as optimization problems.

This thesis will focus on frequency-domain model updating approaches, particularly the modal property difference and modal dynamic residual approaches. As mentioned before, both the modal property difference and modal dynamic residual approach can be formulated as optimization problems. At first, researchers used sensitivity or iterative linearization method to solve those optimization problems [24, 34]. Additionally, those approaches can be potentially solved using commercial optimization software, such as MATLAB optimization toolbox [35]. However, these optimization problems are usually nonconvex with unknown number of local minima. It is generally very difficult, if not impossible, for the optimization solvers to guarantee that the solution is globally optimal. Accordingly, it is not guaranteed that the structural parameter values found by these local optimization algorithms provide the globally smallest possible objective function value.

1.2 Introduction of Optimization Algorithms

When solving a smooth optimization problem whose objective function is differentiable everywhere within the feasible set, gradient based local search algorithms, such as steepest descend, Newton's method, quasi-Newton's method and so forth, are usually selected [36]. The algorithms only guarantee local optimality in the neighborhood

of the final solution, and by no means guarantee that the final solution is minimal in the entire feasible solution space. Figure 1-1 shows a nonconvex function, $f(x)$: $\mathbb{R} \rightarrow \mathbb{R}$, with multiple local minima and one global minimum. It is highly possible that the final solution from the local search optimization algorithm is one of those local minima instead of the global minimum. Although in 1D space, the function can be plotted for viewing, most optimization problems of concern in this research are in high-dimensional space where visualization is not possible.

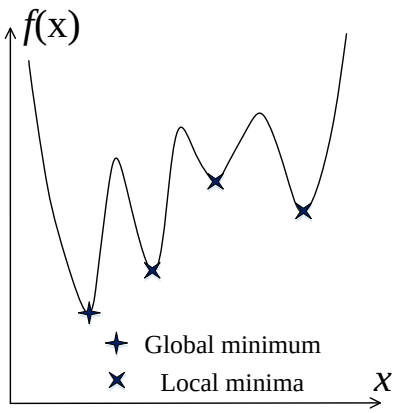


Figure 1-1. Illustration of a nonconvex function

On the other hand, some MATLAB optimization toolbox adopts random search procedures, such as the so-called global search and multistart functions, to find multiple minima of the objective function. A random search procedure first determines multiple sets of initial values of the optimization variables. Starting from each set of initial values, the procedure still utilizes gradient based local search algorithms to find a local minimum. Although the smallest among all identified local minima is selected in the end, the global optimality still cannot be guaranteed. The reason is that the number of local minima and

their neighborhood regions are usually unknown, thus it is impossible to guarantee a starting point has been allocated in the neighborhood of the global minimum.

On the other hand, the gradient-based local search algorithm is not applicable for nonsmooth optimization problems, because that the objective function cannot be differentiated at certain points within the feasible region. Alternatively, other stochastic searching algorithms, such as these mimicking a physical process (e.g. simulated annealing algorithm [37]), a natural selection process (e.g. genetic algorithm [38]), or social behavior of animals (e.g. particle swarm [39]), can be adopted. Researchers also attempt to solve the optimization problems in FE model updating using the stochastic search algorithms [40]. However, those algorithms can only improve the chances of finding a global solution for nonconvex optimization problems, while the global optimality of the solution still cannot be guaranteed.

In order to overcome the limitation of local search algorithms, this thesis will investigate global optimization algorithms that can guarantee the global optimality of the final solution. The first global optimization algorithm to be studied is the branch-and-bound (B&B) algorithm [41]. The algorithm attempts to solve a nonconvex optimization problem by iteratively subdividing the feasible set of the optimization variables and obtaining the upper and lower bounds of the optimal objective function value. The upper bounds are obtained by finding a local minimum within each subset. To obtain a lower bound, convex underestimation of the objective function in each subset has been studied [41]. Furthermore, a branch-and-reduce approach is proposed to accelerate the convergence of the B&B process [42]. More recently, a polyhedral branch-and-cut approach is proposed to further improve the convergence rate of the B&B algorithm [43].

For application of B&B algorithm on bilinear optimization problems, a new relaxation and branching approach is introduced by Dey *et al* [44]. Beside the B&B algorithm, a second global optimization algorithm to be studied is the P-RD algorithm [45-47]. The P-RD algorithm is applicable for solving a particular type of nonconvex optimization problems, which is named as biconvex problems. Benefiting from the biconvex characteristics of the problem, the P-RD algorithm iteratively approximates the lower and upper bounds of the optimal objective function value, and reduces the gap between the lower and upper bounds. When the gap between the lower and upper bound is smaller than the prescribed tolerance value, the P-RD algorithm decides that the optimization process reaches the global optimal value of the optimization problem.

1.3 Organization of Dissertation

The rest of this dissertation is organized as follows.

Chapter 2 introduces three model updating formulations for FE model updating. Two of the formulations belong to the category of modal property difference approach, and the third is the modal dynamic residual approach. Along with each formulation, a detailed description of the Jacobian derivative of the objective function with respect to optimization variables is presented. To solve the optimization problems, two local search optimization algorithms are introduced. The presented model updating formulations with optimization algorithms are evaluated with numerical simulations.

Chapter 3 extends the model updating formulations and local search optimization algorithms to the data collected from laboratory and field experiments. In order to collect dense data from actual structure at low cost, a wireless sensing node, named *Martlet*, is developed. Laboratory and field experiments are conducted to validate the performance of

Martlet wireless sensing node. The modal properties extracted from the collected structural vibration data are fed into the model updating formulations to identify the structural parameters.

Chapter 4 proposes to apply two global optimization algorithms, i.e. B&B and P-RD, to the model updating formulations. The two global optimization algorithms are first introduced. In order to apply the B&B and P-RD algorithms, both modal property difference and modal dynamic residual approaches requires reformulation. Finally, numerical simulation and laboratory experiment are performed to validate the performance of the model updating formulations and global optimization algorithms.

Chapter 5 presents a summary of the research and primary conclusions. Future research topics are recommended.

CHAPTER 2. FE MODEL UPDATING FORMULATIONS AND LOCAL SEARCH OPTIMIZATION ALGORITHMS

As stated in CHAPTER 1, frequency-domain finite element (FE) model updating approaches are usually formulated as mathematical optimization problems. This section first reviews the concepts of convex optimization. The objective functions of finite element (FE) model updating problems are then be presented. In addition, to facilitate the optimization process, the Jacobian derivative of the objective function for each formulation is presented in detail. Two local search optimization algorithms, i.e. Levenberg-Marquardt and trust-region-reflective algorithms, implemented in `lsqnonlin` solver of MATLAB optimization toolbox [35] are adopted to numerically solve the optimization problems. Finally, the presented FE model updating formulations with local search optimization algorithms are validated through numerical studies.

2.1 Convex Optimization

As introduced in CHAPTER 1, the frequency-domain FE model updating approaches can be formulated as optimization problems with selected updating parameters as optimization variables. A general optimization problem with vector variable $\mathbf{x} \in \mathbb{R}^{n_\mathbf{x}}$ can be formulated as below:

$$\underset{\mathbf{x}}{\text{minimize}}\ f(\mathbf{x}) \tag{2-1a}$$

$$\text{subject to } g_i(\mathbf{x}) \leq 0, i = 1 \ldots p \tag{2-1b}$$

$$h_i(\mathbf{x}) = 0, i = 1 \dots q \tag{2-1c}$$

where $f, g_i, i = 1 \dots p$ and $h_i, i = 1 \dots q: \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$ are functions of optimization variables $\mathbf{x}$. However, except when is convex, the optimization problem has unknown number of local minima [36, 48, 49]. For the optimization problem defined in Eq. (2-1) to be convex, the first requirement is that the objective function, f, is a convex function. A function $f: \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$ is convex, if for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom} f$, and any $\theta \in \mathbb{R}$ with $0 \leq \theta \leq 1$, we have:

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}) \tag{2-2}$$

Figure 2-1 illustrates a convex function $f: \mathbb{R} \to \mathbb{R}$. The function value at the interpolation of points x and y, $f(\theta x + (1-\theta)y)$, should be less than or equal to the interpolation of the function values at x and y, $\theta f(x) + (1-\theta) f(y)$.

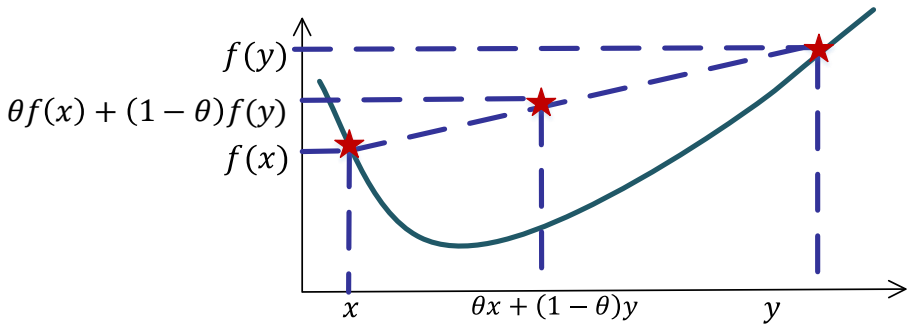


Figure 2-1. Illustration of a convex function

Secondly, for the optimization problem in Eq. (2-1) to be convex, the functions g_i and h_i are required to be convex and affine, respectively. As a result, the feasible set, $\{\mathbf{x} | g_i(\mathbf{x}) \leq 0, i = 1 \dots p; h_i(\mathbf{x}) = 0, i = 1 \dots q\}$, is a convex set. Here a set $\mathbb{C}$ is convex if and only if for any $\mathbf{x}, \mathbf{y} \in \mathbb{C}$ and any $\theta \in \mathbb{R}$ with $0 \leq \theta \leq 1$, we have:

$$\theta \mathbf{x} + (1 - \theta)\mathbf{y} \in \mathbb{C} \tag{2-3}$$

Eq. (2-3) means that if the line segment between any two points in $\mathbb{C}$ always lies in $\mathbb{C}$, $\mathbb{C}$ is a convex set. Figure 2-2 presents a simple example of a convex set, $\mathbb{C} \in \mathbb{R}^2$.

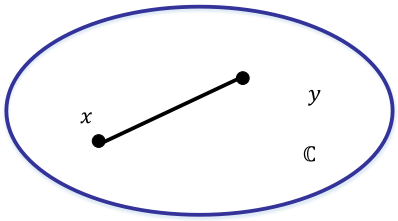


Figure 2-2. Illustration of a convex set

For the optimization problem to be Eq. (2-1) is convex, the convex objective function, $f(\mathbf{x})$, is minimized over a convex set, $\{\mathbf{x}|g_i(\mathbf{x}) \leq 0, i = 1 \dots p; h_i(\mathbf{x}) = 0, i = 1 \dots q\}$. In the following sections, the formulations of a few different FE model updating approaches will be presented. The convexity of the corresponding optimization problems will also be discussed.

2.2 FE Model Updating Formulations

In order to update selected stiffness parameters of a linear structure, a vector variable, $\boldsymbol{\alpha} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$, is formed to contain the corresponding updating variables. The j-th ($j = 1 \dots n_{\boldsymbol{\alpha}}$) entry of $\boldsymbol{\alpha}$, α_j, corresponds to a parameter such as Young's modulus, or support spring stiffness. Each α_j is to be treated as an optimization variable in the optimization problem that attempts to update the FE model closer to the actual structure. In this study, $\boldsymbol{\alpha}$ is scaled to represent the relative change percentage from nominal value of each stiffness parameter. As a result, a value of $\alpha_j = 0$ means the parameter takes the same value as the

nominal one; a value of $\alpha_j = -0.5$ means a 50% reduction from the nominal value. The stiffness matrix can be formulated as an affine matrix function of the vector variable $\boldsymbol{\alpha}$:

$$\mathbf{K}(\boldsymbol{\alpha}) = \mathbf{K_0} + \sum_{j=1}^{n_{\boldsymbol{\alpha}}} \alpha_j \mathbf{K}_j \tag{2-4}$$

Here $\mathbf{K}(\boldsymbol{\alpha}): \mathbb{R}^{n_{\boldsymbol{\alpha}}} \rightarrow \mathbb{R}^{N \times N}$ represents an affine matrix function of $\boldsymbol{\alpha}$; *N* denotes the number of degrees of freedom (DOFs) of the structure; $\mathbf{K}_0$ is the nominal stiffness matrix prior to model updating, usually generated based on design drawings and nominal material properties; $\mathbf{K}_j$ is a constant influence matrix of the *j*-th stiffness parameter being updated, which corresponds to the updating variable α_j. In this study, it is assumed that the structural mass matrix (**M**) is accurate enough and does not require updating. When needed, a similar formulation can be constructed to update mass parameters such as density values.

The FE model updating approaches studied in this research are based on frequency-domain modal properties, i.e. resonance frequencies and mode shapes. Using dynamic testing data collected from an as-built structure, usually the first few resonance frequencies, $\omega_i^{\mathrm{EXP}}, i = 1 \ldots n_{\mathrm{modes}}$, and the corresponding mode shapes can be extracted. Here n_{modes} denotes the number of experimentally measured modes. For each ω_i^{EXP}, an "experimental eigenvalue" is easily calculated as $\lambda_i^{\mathrm{EXP}} = \left(\omega_i^{\mathrm{EXP}}\right)^2$. For each mode shape eigenvector, the experimental data only provides entries for the DOFs with sensor instrumentation, i.e. $\boldsymbol{\psi}_i^{\mathrm{EXP,m}} \in \mathbb{R}^{n_{\mathrm{m}}}$ for the measured DOFs. On the other hand, the simulated modal properties (λ_i and $\boldsymbol{\psi}_i \in \mathbb{R}^N$) can be generated by the FE model. For example, for certain updating variable $\boldsymbol{\alpha}$, the stiffness matrix $\mathbf{K}(\boldsymbol{\alpha})$ is first assembled using Eq. (2-4). The simulated

eigenvalues and eigenvectors, λ_i and $\boldsymbol{\psi}_i$, are the solution of the generalized eigenvalue problem:

$$[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]\{\boldsymbol{\psi}_i\} = \mathbf{0} \tag{2-5}$$

In the sense that λ_i and $\boldsymbol{\psi}_i$ implicitly depend on $\boldsymbol{\alpha}$, they are written as functions of $\boldsymbol{\alpha}$ when applicable, as $\lambda_i(\boldsymbol{\alpha}): \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}$ and $\boldsymbol{\psi}_i(\boldsymbol{\alpha}): \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}^N$. To reflect the measured DOFs in the formulation, define $\boldsymbol{\psi}_i(\boldsymbol{\alpha}) = [\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha}) \quad \boldsymbol{\psi}_i^{\mathrm{u}}(\boldsymbol{\alpha})]^{\mathrm{T}}$, where $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha}) \in \mathbb{R}^{n_{\mathrm{m}}}$ corresponds to DOFs that are measured/instrumented, and $\boldsymbol{\psi}_i^{\mathrm{u}}(\boldsymbol{\alpha}) \in \mathbb{R}^{n_{\mathrm{u}}}$ corresponds to the unmeasured DOFs. Note that $n_{\mathrm{m}} + n_{\mathrm{u}} = N$, the total number of DOFs. The entry in $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ with the largest magnitude is denoted the q_i-th entry ($1 \le q_i \le n_{\mathrm{m}}$), and $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ is normalized so that the q_i-th entry equals 1. Correspondingly, the simulated eigenvector, $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$, is also normalized so that the q_i-th entry equals 1.

2.2.1 Modal Property Difference Formulation with MAC Values

The first objective function of modal property difference formulation studied in this chapter is proposed by Moller and Friberg [13], where the (vector) optimization variable $\boldsymbol{\alpha}$ correspond to stiffness parameters to be updated.

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad \sum_{i=1}^{n_{\text{modes}}} \left\{ \left(\frac{\lambda_i^{\mathrm{EXP}} - \lambda_i(\boldsymbol{\alpha})}{\lambda_i^{\mathrm{EXP}}} \cdot w_{\lambda_i} \right)^2 + \left(\frac{1 - \sqrt{\mathrm{MAC}_i(\boldsymbol{\alpha})}}{\sqrt{\mathrm{MAC}_i(\boldsymbol{\alpha})}} \cdot w_{\boldsymbol{\psi}_i} \right)^2 \right\} \tag{2-6a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\boldsymbol{\alpha}} \le \boldsymbol{\alpha} \le \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{2-6b}$$

where w_{λ_i} represents the weighting factor of the eigenvalue difference; $w_{\boldsymbol{\psi}_i}$ represents the weighting factor of the eigenvector difference; $\boldsymbol{L}_{\boldsymbol{\alpha}}$ and $\boldsymbol{U}_{\boldsymbol{\alpha}} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$ denote the lower and upper bounds for the optimization variable vector $\boldsymbol{\alpha}$, respectively. The sign "≤" is overloaded to represent element-wise inequality; $\mathrm{MAC}_i(\boldsymbol{\alpha})$ represents the modal assurance criterion between the *i*-th experimental and simulated mode shapes/eigenvectors at measured DOFs, i.e. $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ [50].

$$\mathrm{MAC}_i(\boldsymbol{\alpha}) = \frac{\left(\left(\boldsymbol{\psi}_i^{\mathrm{EXP,m}}\right)^{\mathrm{T}} \boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)^2}{\left\|\boldsymbol{\psi}_i^{\mathrm{EXP,m}}\right\|_2^2 \left\|\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right\|_2^2}, i = 1 \dots n_{\mathrm{modes}} \tag{2-7}$$

Here $\|\cdot\|_2$ denotes the $\mathcal{L}_2$-norm of a vector. Ranging from 0 to 1, the MAC value represents the similarity between two vectors. When two vectors are collinear, the MAC value is close to 1. When two vectors are orthogonal, the MAC value is close to 0. Normally, the numerical optimization algorithms are iterative. At every iteration step, the algorithm recalculates the value of the objective function, using the updated value of $\boldsymbol{\alpha}$ at current step. Specifically, the stiffness matrix $\mathbf{K}(\boldsymbol{\alpha})$ is first assembled using the current value of $\boldsymbol{\alpha}$ (Eq. (2-4)). Then, the simulated modal properties, i.e. $\lambda_i(\boldsymbol{\alpha})$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$, are obtained by solving the generalized eigenvalue problem using the new matrix $\mathbf{K}(\boldsymbol{\alpha})$, as shown in Eq.(2-5). Finally, an updated set of simulated eigenvalues and eigenvectors from FE model are used to evaluate the objective function in Eq. (2-6a). Using nomenclature in optimization, the objective function is an oracle form of updating variable $\boldsymbol{\alpha}$, where the function can be evaluated for any feasible $\boldsymbol{\alpha}$, but has no explicit form. This objective function is generally nonconvex and has unknown number of local minima [36]. As a

result, off-the-shelf local search algorithms usually cannot guarantee the global optimality. To increase the chance of finding the global minimum, optimization process can be initiated from multiple starting values randomized within the bounds.

2.2.2 Modal Property Difference Formulation with Eigenvector Difference

Still using $\boldsymbol{\alpha}$ as the optimization variables, the second objective function of modal property difference formulation is directly based upon the differences between the mode shape/eigenvector entries at the measured DOFs.

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad \sum_{i=1}^{n_{\text{modes}}} \left\{ \left(\frac{\lambda_i^{\text{EXP}} - \lambda_i(\boldsymbol{\alpha})}{\lambda_i^{\text{EXP}}} \cdot w_{\lambda_i} \right)^2 + \left\| \mathbf{Q}_i \left\{ \boldsymbol{\psi}_i^{\text{EXP,m}} - \boldsymbol{\psi}_i^{\text{m}}(\boldsymbol{\alpha}) \right\} \cdot w_{\boldsymbol{\psi}_i} \right\|_2^2 \right\} \tag{2-8a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{2-8b}$$

here the selection matrix $\mathbf{Q}_i \in \mathbb{R}^{(n_{\text{m}}-1)\times n_{\text{m}}}$ is defined as

$$\mathbf{Q}_i = \begin{bmatrix} \mathbf{I}_{q_i-1} & \mathbf{0}_{(q_i-1)\times 1} & \mathbf{0}_{(q_i-1)\times(n_{\text{m}}-q_i)} \\ \mathbf{0}_{(n_{\text{m}}-q_i)\times(q_i-1)} & \mathbf{0}_{(n_{\text{m}}-q_i)\times 1} & \mathbf{I}_{n_{\text{m}}-q_i} \end{bmatrix} \tag{2-9}$$

where $\mathbf{I}_{q_i-1}$ and $\mathbf{I}_{n_{\text{m}}-q_i}$ denote identity matrices with size of $q_i - 1$ and $n_{\text{m}} - q_i$, respectively. Recall that q_i is the entry in $\boldsymbol{\psi}_i^{\text{EXP,m}}$ and $\boldsymbol{\psi}_i^{\text{m}}(\boldsymbol{\alpha})$ that equals 1, i.e. $\psi_{q_i,i}^{\text{EXP,m}} = \psi_{q_i,i}^{\text{m}}(\boldsymbol{\alpha}) = 1$. It can be seen that upon the aforementioned normalization of $\boldsymbol{\psi}_i^{\text{EXP,m}}$ and $\boldsymbol{\psi}_i^{\text{m}}(\boldsymbol{\alpha})$, Eq. (2-8a) directly minimizes the differences between the entries in the experimental and simulated eigenvectors at measured DOFs (except for the q_i-th entry). This is different from using MAC values to quantify the vector difference as shown in Eq.

(2-6a). Similar to Eq. (2-6a), the objective function in Eq. (2-8a) is also in oracle formulation of updating variable $\boldsymbol{\alpha}$. The formulation is generally nonconvex as well. As a result, with unknown number of local minima, local search algorithms usually cannot guarantee the global optimality.

2.2.3 Modal Dynamic Residual Formulation

In comparison with the modal property difference formulations, the modal dynamic residual formulation attempts to minimize the residuals of the generalized eigenvalue equations, as shown in Eq. (2-10). Matrices given by the FE model are used in combination with experimentally-measured modal properties for calculating the modal dynamic residuals. The sum of the residual squares forms the objective function. We define $\boldsymbol{\varphi}^{\mathrm{u}} = [\boldsymbol{\psi}_1^{\mathrm{u}} \quad \boldsymbol{\psi}_2^{\mathrm{u}} \quad \cdots \quad \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{u}}]^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{u}} \cdot n_{\mathrm{modes}}}$ that corresponds to the unmeasured DOFs of the simulated eigenvector. In the optimization problem of the modal dynamic residual formulation, both $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$ are treated as the optimization variables:

$$\underset{\boldsymbol{\alpha},\boldsymbol{\varphi}^{\mathrm{u}}}{\text{minimize}} \sum_{i=1}^{n_{\mathrm{modes}}} \left\| \left[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}}\mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_i^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_i^{\mathrm{u}} \end{Bmatrix} \cdot w_i \right\|_2^2 \tag{2-10a}$$

$$\text{subject to } \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{2-10b}$$

$$\boldsymbol{L}_{\boldsymbol{\varphi}^{\mathrm{u}}} \leq \boldsymbol{\varphi}^{\mathrm{u}} \leq \boldsymbol{U}_{\boldsymbol{\varphi}^{\mathrm{u}}} \tag{2-10c}$$

Here w_i represents the weighting factor for the i-th modal residuals; $\boldsymbol{L}_{\boldsymbol{\varphi}^{\mathrm{u}}}$ and $\boldsymbol{U}_{\boldsymbol{\varphi}^{\mathrm{u}}} \in \mathbb{R}^{n_{\mathrm{u}} \cdot n_{\mathrm{modes}}}$ denote the lower and upper bounds for the optimization variable vector $\boldsymbol{\varphi}^{\mathrm{u}}$, respectively. Similar to the modal property difference formulations, in order to evaluate

the objective function value at an iteration step, the stiffness matrix, $\mathbf{K}(\boldsymbol{\alpha})$ is first assembled using the current value of $\boldsymbol{\alpha}$ (Eq. (2-4)). Combined with the experimental eigenvalues λ_i^{EXP}, experimental eigenvectors at measured DOFs $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$, as well as the current values of eigenvectors at unmeasured DOFs ($\boldsymbol{\varphi}^{\mathrm{u}}$), the objective function in Eq. (2-10a) is evaluated. The feasible regions of the optimization variables, $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$, are both convex box constraint. The objective function of the modal dynamic residual approach is nonconvex with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$ because of the matrix-vector multiplication inside the $\mathcal{L}_2$-norm function. The matrix contains variable $\boldsymbol{\alpha}$ and the vector contains variable $\boldsymbol{\varphi}^{\mathrm{u}}$. Therefore, the optimization problem of the modal dynamic residual formulation is also nonconvex. However, unlike the modal property difference formulations, the optimization problem shown in Eq.(2-10) is no longer in oracle form of the optimization variables.

2.3 Jacobian Derivative of Model Updating Formulations

As mentioned in Section 2.2, the numerical optimization algorithms solving Eqs. (2-6), (2-8), and (2-10) are iterative. At every iteration step, the Jacobian derivative (short-named as Jacobian) of the objective function is often used to determine the search direction in gradient based local search algorithms for smooth optimization problems. For an objective function $f(\mathbf{x})\colon \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}$, the Jacobian is defined as $\mathrm{D}_{\mathbf{x}} f = \left[\frac{\partial f}{\partial x_1} \quad \frac{\partial f}{\partial x_2} \quad \cdots \quad \frac{\partial f}{\partial x_{n_{\mathbf{x}}}}\right] \in \mathbb{R}^{1\times n_{\mathbf{x}}}$. In this subsection, the Jacobian of the objective function shown in Eqs. (2-6), (2-8), and (2-10) will the described in detail.

2.3.1 Jacobian of MAC Value Formulation

To facilitate the Jacobian derivation, the objective function in Eq. (2-6) needs some rewriting. A residual vector function, $\mathbf{r}(\boldsymbol{\alpha})\colon \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}^{2\cdot n_{\text{modes}}}$ is first defined.

$$\mathbf{r}(\boldsymbol{\alpha}) = \begin{bmatrix} \mathbf{r}_1(\boldsymbol{\alpha}) \\ \mathbf{r}_2(\boldsymbol{\alpha}) \\ \vdots \\ \mathbf{r}_{n_{\text{modes}}}(\boldsymbol{\alpha}) \end{bmatrix} \tag{2-11}$$

where $\mathbf{r}_i(\boldsymbol{\alpha})\colon \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}^2$ is shown below:

$$\mathbf{r}_i(\boldsymbol{\alpha}) = \begin{bmatrix} \dfrac{\lambda_i^{\text{EXP}} - \lambda_i(\boldsymbol{\alpha})}{\lambda_i^{\text{EXP}}} \cdot w_{\lambda_i} \\ \dfrac{1 - \sqrt{\text{MAC}_i(\boldsymbol{\alpha})}}{\sqrt{\text{MAC}_i(\boldsymbol{\alpha})}} \cdot w_{\boldsymbol{\psi}_i} \end{bmatrix}, i = 1 \dots n_{\text{modes}} \tag{2-12}$$

Using $\mathbf{r}(\boldsymbol{\alpha})$, the optimization problem in Eq. (2-6) is equivalent to

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad f(\boldsymbol{\alpha}) = \mathbf{r}(\boldsymbol{\alpha})^{\mathbf{T}}\mathbf{r}(\boldsymbol{\alpha}) \tag{2-13a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{2-13b}$$

The Jacobian for $f(\boldsymbol{\alpha})$ in Eq. (2-13a), $\mathrm{D}_{\boldsymbol{\alpha}} f \in \mathbb{R}^{1\times n_{\boldsymbol{\alpha}}}$, equals $\mathrm{D}_{\mathbf{r}} f \cdot \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}$ by using the chain rule. The first entry is $\mathrm{D}_{\mathbf{r}} f = 2\mathbf{r}^{\mathrm{T}} \in \mathbb{R}^{1\times(2\cdot n_{\text{modes}})}$. The second entry is $\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r} = [\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_1 \quad \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_2 \quad \cdots \quad \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_{n_{\text{modes}}}]^{\mathrm{T}} \in \mathbb{R}^{(2\cdot n_{\text{modes}})\times n_{\boldsymbol{\alpha}}}$. Recall the definition of MAC value in Eq. (2-7), each $\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_i \in \mathbb{R}^{2\times n_{\boldsymbol{\alpha}}}$ can be formed as:

$$\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_i = \begin{bmatrix} -\frac{w_{\lambda_i}}{\lambda_i^{\mathrm{EXP}}} \cdot \mathrm{D}_{\boldsymbol{\alpha}}\left(\lambda_i(\boldsymbol{\alpha})\right) \\ \frac{-w_{\boldsymbol{\psi}_i}}{\sqrt{\mathrm{MAC}_i(\boldsymbol{\alpha})}} \cdot \left(\frac{\left(\boldsymbol{\psi}_i^{\mathrm{EXP,m}}\right)^{\mathrm{T}}}{\left(\boldsymbol{\psi}_i^{\mathrm{EXP,m}}\right)^{\mathrm{T}} \boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})} - \frac{\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)^{\mathrm{T}}}{\left\|\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right\|_2^2} \right) \mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right) \end{bmatrix}, i = 1 \dots n_{\mathrm{modes}} \tag{2-14}$$

The formulation for $\mathrm{D}_{\boldsymbol{\alpha}}\left(\lambda_i(\boldsymbol{\alpha})\right) \in \mathbb{R}^{1 \times n_{\boldsymbol{\alpha}}}$ and $\mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right) \in \mathbb{R}^{n_{\mathrm{m}} \times n_{\boldsymbol{\alpha}}}$ have been well studied by researchers [51, 52]. The process shown below of obtaining $\mathrm{D}_{\boldsymbol{\alpha}}\left(\lambda_i(\boldsymbol{\alpha})\right)$ and $\mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)$ is similar to the one described in [53]. Nevertheless, a simplified way of obtaining $\mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)$ based on the normalization of $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ is presented, without expressing the derivative as a linear combination of all the eigenvectors (as in [52]). Recall the generalized eigenvalue equation for the *i*-th mode:

$$[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]\{\boldsymbol{\psi}_i\} = \mathbf{0} \tag{2-15}$$

By differentiating Eq. (2-15) with respect to the *j*-th updating variable, α_j, the following equation can be obtained.

$$[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}] \frac{\partial \boldsymbol{\psi}_i}{\partial \alpha_j} = \frac{\partial \lambda_i}{\partial \alpha_j} \mathbf{M} \boldsymbol{\psi}_i - \mathbf{K}_j \boldsymbol{\psi}_i \tag{2-16}$$

Assume that the eigenvalues are distinct, and define the modal mass of the *i*-th mode as $m_i = (\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{M} \boldsymbol{\psi}_i$. Pre-multiplying Eq. (2-16) by $(\boldsymbol{\psi}_i)^{\mathrm{T}}$and noting $(\boldsymbol{\psi}_i)^{\mathrm{T}}[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}] = \mathbf{0}$, Eq. (2-16) can be simplified as follows:

$$(\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{K}_j \boldsymbol{\psi}_i = \frac{\partial \lambda_i}{\partial \alpha_j} (\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{M} \boldsymbol{\psi}_i \tag{2-17a}$$

$$\frac{\partial \lambda_i}{\partial \alpha_j} = \frac{(\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{K}_j \boldsymbol{\psi}_i}{m_i} \tag{2-17b}$$

As a result, Jacobian of the i-th simulated eigenvalue, $\mathrm{D}_{\boldsymbol{\alpha}}(\lambda_i) \in \mathbb{R}^{1 \times n_{\boldsymbol{\alpha}}}$, with respect to updating vector, $\boldsymbol{\alpha}$, can be found as follows:

$$\begin{aligned} \mathrm{D}_{\boldsymbol{\alpha}}(\lambda_i) &= \left[\frac{\partial \lambda_i}{\partial \alpha_1} \quad \frac{\partial \lambda_i}{\partial \alpha_2} \quad \cdots \quad \frac{\partial \lambda_i}{\partial \alpha_{n_{\boldsymbol{\alpha}}}} \right] \\ &= \left[\frac{(\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{K}_1 \boldsymbol{\psi}_i}{m_i} \quad \frac{(\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{K}_2 \boldsymbol{\psi}_i}{m_i} \quad \cdots \quad \frac{(\boldsymbol{\psi}_i)^{\mathrm{T}} \mathbf{K}_{n_{\boldsymbol{\alpha}}} \boldsymbol{\psi}_i}{m_i} \right] \end{aligned} \tag{2-18}$$

After obtaining $\partial \lambda_i / \partial \alpha_j$, Eq. (2-16) is reused to find the only remaining unknown term, $\partial \boldsymbol{\psi}_i / \partial \alpha_j \in \mathbb{R}^N$. However, $\partial \boldsymbol{\psi}_i / \partial \alpha_j$ cannot be directly obtained from Eq. (2-16), because $[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]$ is rank deficient by one assuming that the eigenvalue λ_i is distinct. Nevertheless, as previously mentioned, $\boldsymbol{\psi}_i^{\mathrm{m}}$ is normalized so that the q_i-th entry always equals a constant 1. As a result, the q_i-th entry in vector $\partial \boldsymbol{\psi}_i^{\mathrm{m}} / \partial \alpha_j$ is zero, i.e. $\partial \psi_{q_i,i}^{\mathrm{m}} / \partial \alpha_j = 0$. Because of the separation by measured and unmeasured DOFs, $\boldsymbol{\psi}_i = [\boldsymbol{\psi}_i^{\mathrm{m}} \quad \boldsymbol{\psi}_i^{\mathrm{u}}]^{\mathrm{T}}$, the q_i-th entry in $\partial \boldsymbol{\psi}_i / \partial \alpha_j$ is also zero, i.e. $\partial \psi_{q_i,i} / \partial \alpha_j = 0$. This is utilized to resolve the rank deficiency issue of $[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]$. Specifically, define $\mathbf{P}_i = \begin{bmatrix} \mathbf{Q}_i & \mathbf{0}_{(n_{\mathrm{m}}-1) \times n_{\mathrm{u}}} \\ \mathbf{0}_{n_{\mathrm{u}} \times n_{\mathrm{m}}} & \mathbf{I}_{n_{\mathrm{u}}} \end{bmatrix} \in \mathbb{R}^{(N-1) \times N}$, which extends $\mathbf{Q}_i$ in Eq. (2-9) from measured DOFs to all DOFs. Then, pre-multiplying and post-multiplying $[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}]$ in Eq. (2-16) by $\mathbf{P}_i$ and $\mathbf{P}_i^{\mathrm{T}}$ to cross out the q_i-th row and q_i-th column, $\mathbf{B}_i \in \mathbb{R}^{(N-1) \times (N-1)}$ is generated.

$$\mathbf{B}_i = \mathbf{P}_i [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}] \mathbf{P}_i^{\mathrm{T}} \tag{2-19}$$

Next, pre-multiply $\left(\frac{\partial \lambda_i}{\partial \alpha_j}\mathbf{M}\boldsymbol{\psi}_i - \mathbf{K}_j\boldsymbol{\psi}_i\right)$ in Eq. (2-16) by $\mathbf{P}_i$ to eliminate the q_i-th row and obtain $\mathbf{b}_{ij} \in \mathbb{R}^{N-1}$:

$$\mathbf{b}_{ij} = \mathbf{P}_i \cdot \left(\frac{\partial \lambda_i}{\partial \alpha_j}\mathbf{M}\boldsymbol{\psi}_i - \mathbf{K}_j\boldsymbol{\psi}_i\right) \tag{2-20}$$

Finally, recalling $\partial \psi_{q_i,i}/\partial \alpha_j = 0$, the elimination of the q_i-th row in Eq. (2-16) is equivalent to the following.

$$\mathbf{B}_i \begin{Bmatrix} \dfrac{\partial(\mathbf{Q}_i\boldsymbol{\psi}_i^{\mathrm{m}})}{\partial \alpha_j} \\ \dfrac{\partial(\boldsymbol{\psi}_i^{\mathrm{u}})}{\partial \alpha_j} \end{Bmatrix} = \mathbf{b}_{ij} \tag{2-21}$$

Thus, the Jacobian of the *i*-th simulated eigenvector can be shown as:

$$\begin{Bmatrix} \dfrac{\partial(\mathbf{Q}_i\boldsymbol{\psi}_i^{\mathrm{m}})}{\partial \alpha_j} \\ \dfrac{\partial(\boldsymbol{\psi}_i^{\mathrm{u}})}{\partial \alpha_j} \end{Bmatrix} = \mathbf{B}_i^{-1}\mathbf{b}_{ij} \tag{2-22}$$

In summary, $\partial \boldsymbol{\psi}_i^{\mathrm{m}}/\partial \alpha_j$ has 0 at the q_i-th entry and other entries are provided by the equation above. The Jacobian of the *i*-th simulated eigenvector at measured DOFs, $\mathrm{D}_{\boldsymbol{\alpha}}(\boldsymbol{\psi}_i^{\mathrm{m}}) \in \mathbb{R}^{n_{\mathrm{m}} \times n_{\boldsymbol{\alpha}}}$ in Eq. (2-14), can be obtained as follows:

$$\mathbf{D}_{\boldsymbol{\alpha}}(\boldsymbol{\psi}_i^{\mathbf{m}}) = \begin{bmatrix} \dfrac{\partial \boldsymbol{\psi}_i^{\mathrm{m}}}{\partial \alpha_1} & \dfrac{\partial \boldsymbol{\psi}_i^{\mathrm{m}}}{\partial \alpha_2} & \cdots & \dfrac{\partial \boldsymbol{\psi}_i^{\mathrm{m}}}{\partial \alpha_{n_{\boldsymbol{\alpha}}}} \end{bmatrix} \tag{2-23}$$

$$= \begin{bmatrix} \partial\psi_{1,i}^{\mathrm{m}}/\partial\alpha_1 & \partial\psi_{1,i}^{\mathrm{m}}/\partial\alpha_2 & \cdots & \partial\psi_{1,i}^{\mathrm{m}}/\partial\alpha_{n_{\boldsymbol{\alpha}}} \\ \vdots & \vdots & \cdots & \vdots \\ \partial\psi_{q_i-1,i}^{\mathrm{m}}/\partial\alpha_1 & \partial\psi_{q_i-1,i}^{\mathrm{m}}/\partial\alpha_2 & \cdots & \partial\psi_{q_i-1,i}^{\mathrm{m}}/\partial\alpha_{n_{\boldsymbol{\alpha}}} \\ 0 & 0 & \cdots & 0 \\ \psi_{q_i+1,i}^{\mathrm{m}}/\partial\alpha_1 & \partial\psi_{q_i+1,i}^{\mathrm{m}}/\partial\alpha_2 & \cdots & \partial\psi_{q_i+1,i}^{\mathrm{m}}/\partial\alpha_{n_{\boldsymbol{\alpha}}} \\ \vdots & \vdots & \cdots & \vdots \\ \psi_{n_{\mathrm{m}},i}^{\mathrm{m}}/\partial\alpha_1 & \psi_{n_{\mathrm{m}},i}^{\mathrm{m}}/\partial\alpha_2 & \cdots & \psi_{n_{\mathrm{m}},i}^{\mathrm{m}}/\partial\alpha_{n_{\boldsymbol{\alpha}}} \end{bmatrix} \begin{matrix} 1 \\ \vdots \\ q_i - 1 \\ q_i \\ q_i + 1 \\ \vdots \\ n_{\mathrm{m}} \end{matrix}$$

After obtaining the Jacobian of simulated eigenvalue and eigenvector at measured DOFs, $\mathrm{D}_{\boldsymbol{\alpha}}(\lambda_i)$ and $\mathrm{D}_{\boldsymbol{\alpha}}(\boldsymbol{\psi}_i^{\mathrm{m}})$, the analytical Jacobian in Eq. (2-14) can be calculated.

2.3.2 Jacobian of Eigenvector Difference Formulation

Similar to the method introduced in Section 2.3.1, in order to derive the Jacobian of the eigenvector difference formulation in Eq. (2-8), a residual vector, $\mathbf{r}(\boldsymbol{\alpha}) : \mathbb{R}^{n_{\boldsymbol{\alpha}}} \rightarrow \mathbb{R}^{n_{\mathrm{m}} \cdot n_{\mathrm{modes}}}$ is defined as follows

$$\mathbf{r}(\boldsymbol{\alpha}) = [\mathbf{r}_1(\boldsymbol{\alpha}) \quad \mathbf{r}_2(\boldsymbol{\alpha}) \quad \ldots \quad \mathbf{r}_{n_{\mathrm{modes}}}(\boldsymbol{\alpha})]^{\mathrm{T}} \tag{2-24}$$

where $\mathbf{r}_i(\boldsymbol{\alpha}) : \mathbb{R}^{n_{\boldsymbol{\alpha}}} \rightarrow \mathbb{R}^{n_{\mathrm{m}}}$ is redefined as follows

$$\mathbf{r}_i(\boldsymbol{\alpha}) = \begin{bmatrix} \dfrac{\lambda_i^{\mathrm{EXP}} - \lambda_i(\boldsymbol{\alpha})}{\lambda_i^{\mathrm{EXP}}} \cdot w_{\lambda_i} \\ \mathbf{Q}_i\left\{\boldsymbol{\psi}_i^{\mathrm{EXP,m}} - \boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right\} \cdot w_{\boldsymbol{\psi}_i} \end{bmatrix}, i = 1 \ldots n_{\mathrm{modes}} \tag{2-25}$$

Using $\mathbf{r}(\boldsymbol{\alpha})$, the optimization problem in Eq. (2-8) for the eigenvector difference formulation can also be rewritten the same as Eq. (2-13) (for the MAC value formulation), with the objective function $f(\boldsymbol{\alpha}) = \mathbf{r}(\boldsymbol{\alpha})^{\mathbf{T}}\mathbf{r}(\boldsymbol{\alpha})$. Again, the Jacobian for $f(\boldsymbol{\alpha})$, $\mathrm{D}_{\boldsymbol{\alpha}}f \in \mathbb{R}^{1 \times n_{\boldsymbol{\alpha}}}$ equals $\mathrm{D}_{\mathbf{r}}f \cdot \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}$ from the chain rule. However, the residual vector $\mathbf{r}$ has a different dimension from Eq. (2-13). For the eigenvector difference formulation, the first Jacobian

entry is $\mathrm{D}_{\mathbf{r}} f = 2\mathbf{r}^{\mathrm{T}} \in \mathbb{R}^{1\times(n_{\mathrm{m}}\cdot n_{\mathrm{modes}})}$. Meanwhile, the second entry is $\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r} = [\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_1 \quad \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_2 \quad \cdots \quad \mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_{n_{\mathrm{modes}}}]^{\mathrm{T}} \in \mathbb{R}^{(n_{\mathrm{m}}\cdot n_{\mathrm{modes}})\times n_{\boldsymbol{\alpha}}}$, where each $\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_i \in \mathbb{R}^{n_{\mathrm{m}}\times n_{\boldsymbol{\alpha}}}$ can be formed as follows:

$$\mathrm{D}_{\boldsymbol{\alpha}}\mathbf{r}_i = \begin{bmatrix} -\dfrac{\mathrm{D}_{\boldsymbol{\alpha}}\left(\lambda_i(\boldsymbol{\alpha})\right)}{\lambda_i^{\mathrm{EXP}}}\cdot w_{\lambda_i} \\ -\mathbf{Q}_i\mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)\cdot w_{\boldsymbol{\psi}_i} \end{bmatrix}, i = 1 \ldots n_{\mathrm{modes}} \tag{2-26}$$

The Jacobian of the *i*-th simulated eigenvalue and eigenvector at measured DOF, $\mathrm{D}_{\boldsymbol{\alpha}}\left(\lambda_i(\boldsymbol{\alpha})\right)$ and $\mathrm{D}_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right)$, have been introduced in Eqs. (2-18) and (2-23).

2.3.3 Jacobian of Modal Dynamic Residual Formulation

As mentioned in Section 2.2.3, the optimization problem of the modal dynamic residual formulation is no longer in an oracle form of its optimization variables. Thus, the Jacobian of the modal dynamic residual formulation is relatively easy to derive. Again, we start with defining a residual vector, $\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) : \mathbb{R}^{n_{\boldsymbol{\alpha}}+n_{\mathrm{u}}\cdot n_{\mathrm{modes}}} \to \mathbb{R}^N$:

$$\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) = \begin{bmatrix} \mathbf{r}_1(\boldsymbol{\alpha}, \boldsymbol{\psi}_1^{\mathrm{u}}) \\ \mathbf{r}_2(\boldsymbol{\alpha}, \boldsymbol{\psi}_2^{\mathrm{u}}) \\ \vdots \\ \mathbf{r}_{n_{\mathrm{modes}}}\left(\boldsymbol{\alpha}, \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{u}}\right) \end{bmatrix} \tag{2-27}$$

where $\mathbf{r}_i(\boldsymbol{\alpha}, \boldsymbol{\psi}_i^{\mathrm{u}}): \mathbb{R}^{n_{\boldsymbol{\alpha}}+n_{\mathrm{u}}} \to \mathbb{R}^N$ is shown as below:

$$\mathbf{r}_i(\boldsymbol{\alpha}, \boldsymbol{\psi}_i^{\mathrm{u}}) = \left[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}}\mathbf{M}\right]\begin{Bmatrix} \boldsymbol{\psi}_i^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_i^{\mathrm{u}} \end{Bmatrix}\cdot w_i, i = 1 \ldots n_{\mathrm{modes}} \tag{2-28}$$

Using $\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}})$, the optimization problem in Eq. (2-10) for the modal dynamic residual formulation can be written as below:

$$\underset{\boldsymbol{\alpha},\boldsymbol{\varphi}^{\mathrm{u}}}{\text{minimize}} \quad f(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) = \mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}})^{\mathrm{T}}\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) \tag{2-29a}$$

$$\text{subject to } \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{2-29b}$$

$$\boldsymbol{L}_{\boldsymbol{\varphi}^{\mathrm{u}}} \leq \boldsymbol{\varphi}^{\mathrm{u}} \leq \boldsymbol{U}_{\boldsymbol{\varphi}^{\mathrm{u}}} \tag{2-29c}$$

The Jacobian of Eq. (2-29) can be obtained with the chain rule by defining an intermediate vector variable, $\boldsymbol{\chi} = \begin{Bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\varphi}^{\mathrm{u}} \end{Bmatrix} \in \mathbb{R}^{n_{\boldsymbol{\alpha}} + n_{\mathrm{u}} \cdot n_{\mathrm{modes}}}$. The Jacobian of the objective function $f(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}})$ with respect to the intermediate vector variable, $\boldsymbol{\chi}$, can be expressed as $\mathrm{D}_{\boldsymbol{\chi}} f = \mathrm{D}_{\mathbf{r}} f \cdot \mathrm{D}_{\boldsymbol{\chi}} \mathbf{r}$, where $\mathrm{D}_{\mathbf{r}} f$ simply equals $2\mathbf{r}^{\mathrm{T}} \in \mathbb{R}^{1 \times (N \cdot n_{\mathrm{modes}})}$ and the Jacobian of residual vector $\mathbf{r}$ with respect to intermediate vector $\boldsymbol{\chi}$ is $\mathrm{D}_{\boldsymbol{\chi}} \mathbf{r} = [\mathrm{D}_{\boldsymbol{\alpha}} \mathbf{r} \quad \mathrm{D}_{\boldsymbol{\varphi}^{\mathrm{u}}} \mathbf{r}] \in \mathbb{R}^{(N \cdot n_{\mathrm{modes}}) \times (n_{\boldsymbol{\alpha}} + n_{\mathrm{u}} \cdot n_{\mathrm{modes}})}$. The first term $\mathrm{D}_{\boldsymbol{\alpha}} \mathbf{r} \in \mathbb{R}^{(N \cdot n_{\mathrm{modes}}) \times n_{\boldsymbol{\alpha}}}$, is found as:

$$\mathrm{D}_{\boldsymbol{\alpha}} \mathbf{r} = \begin{bmatrix} w_1 \mathbf{K}_1 \boldsymbol{\psi}_1^{\mathrm{mix}} & w_1 \mathbf{K}_2 \boldsymbol{\psi}_1^{\mathrm{mix}} & \cdots & w_1 \mathbf{K}_{n_{\boldsymbol{\alpha}}} \boldsymbol{\psi}_1^{\mathrm{mix}} \\ w_2 \mathbf{K}_1 \boldsymbol{\psi}_2^{\mathrm{mix}} & w_2 \mathbf{K}_2 \boldsymbol{\psi}_2^{\mathrm{mix}} & \cdots & w_2 \mathbf{K}_{n_{\boldsymbol{\alpha}}} \boldsymbol{\psi}_2^{\mathrm{mix}} \\ \vdots & \vdots & \cdots & \vdots \\ w_{n_{\mathrm{modes}}} \mathbf{K}_1 \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{mix}} & w_{n_{\mathrm{modes}}} \mathbf{K}_2 \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{mix}} & \cdots & w_{n_{\mathrm{modes}}} \mathbf{K}_{n_{\boldsymbol{\alpha}}} \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{mix}} \end{bmatrix} \tag{2-30}$$

where $\boldsymbol{\psi}_i^{\mathrm{mix}} = \begin{Bmatrix} \boldsymbol{\psi}_i^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_i^{\mathrm{u}} \end{Bmatrix}$. In addition, subdividing the stiffness and mass matrices by columns according to the measured and unmeasured DOFs, i.e. $\mathbf{K}(\boldsymbol{\alpha}) = [\mathbf{K}^{\mathrm{m}}(\boldsymbol{\alpha}) \quad \mathbf{K}^{\mathrm{u}}(\boldsymbol{\alpha})]$ and $\mathbf{M} = [\mathbf{M}^{\mathrm{m}} \quad \mathbf{M}^{\mathrm{u}}]$, the residual vector $\mathbf{r}_i(\boldsymbol{\alpha}, \boldsymbol{\psi}_i^{\mathrm{u}})$ in Eq. (2-28) can be rewritten as follows:

$$\mathbf{r}_i(\boldsymbol{\alpha}, \boldsymbol{\psi}_i^{\mathrm{u}}) = \left[\mathbf{K}^{\mathrm{m}}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}} \mathbf{M}^{\mathrm{m}}\right] \boldsymbol{\psi}_i^{\mathrm{EXP,m}} \cdot w_i + \left[\mathbf{K}^{\mathrm{u}}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}} \mathbf{M}^{\mathrm{u}}\right] \boldsymbol{\psi}_i^{\mathrm{u}} \cdot w_i, i = 1 \ldots n_{\mathrm{modes}} \tag{2-31}$$

Thus, the second term in $\mathrm{D}_{\boldsymbol{\chi}}\mathbf{r}$, $\mathrm{D}_{\boldsymbol{\varphi}^{\mathrm{u}}}\mathbf{r} \in \mathbb{R}^{(N\cdot n_{\mathrm{modes}})\times(n_{\mathrm{u}}\cdot n_{\mathrm{modes}})}$, is found to be a block-diagonal matrix.

$$\mathrm{D}_{\boldsymbol{\varphi}^{\mathrm{u}}}\mathbf{r} = \begin{bmatrix} w_1(\mathbf{K}^{\mathrm{u}}(\boldsymbol{\alpha}) - \lambda_1^{\mathrm{EXP}}\mathbf{M}^{\mathrm{u}}) & 0 & \cdots & 0 \\ 0 & w_2(\mathbf{K}^{\mathrm{u}}(\boldsymbol{\alpha}) - \lambda_2^{\mathrm{EXP}}\mathbf{M}^{\mathrm{u}}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_{n_{\mathrm{modes}}}\left(\mathbf{K}^{\mathrm{u}}(\boldsymbol{\alpha}) - \lambda_{n_{\mathrm{modes}}}^{\mathrm{EXP}}\mathbf{M}^{\mathrm{u}}\right) \end{bmatrix} \tag{2-32}$$

2.4 Local Search Optimization Algorithms

Thus far, both modal property difference and modal dynamic residual formulations are presented as an optimization problem. A number of optimization algorithms can be attempted towards solving the optimization problems. For example, MATLAB optimization toolbox supports various algorithms. However, because the optimization problems in Eqs. (2-6), (2-8) and (2-10) are nonconvex and has unknown number of local minima, these off-the-shelf algorithms can only find local minima. While some can better the chance, none can guarantee the global optimality of the solution [36, 48, 49].

In this research, since the analytical Jacobian of three model updating formulations are available, gradient based local search algorithms are adopted to solve the optimization problems. In this chapter, the `lsqnonlin` solver in MATLAB optimization toolbox [35] is adopted to numerically solve the optimization problems. The solver specializes on nonlinear least squares problems where the objective is to minimize the square of $\mathcal{L}_2$-norm of a residual vector $\mathbf{r} \in \mathbb{R}^{\mathrm{m}}$:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) = \sum_{i=1}^{m} r_i^2(\mathbf{x}) = \|\mathbf{r}(\mathbf{x})\|_2^2 = \mathbf{r}(\mathbf{x})^{\mathrm{T}}\,\mathbf{r}(\mathbf{x}) \tag{2-33}$$

Here $r_i(\mathbf{x}): \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$ as a residual function is usually nonlinear. Assembling the residuals in a vector form, $\mathbf{r}(\mathbf{x}): \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}^m$ is defined as $[r_1(\mathbf{x}) \quad r_2(\mathbf{x}) \quad \cdots \quad r_m(\mathbf{x})]^{\mathrm{T}}$. The gradient, $\nabla f(\mathbf{x}) \in \mathbb{R}^{n_\mathbf{x}}$ and Hessian, $\nabla^2 f(\mathbf{x}) \in \mathbb{R}^{n_\mathbf{x} \times n_\mathbf{x}}$ of $f(\mathbf{x}): \mathbb{R}^{n_\mathbf{x}} \to \mathbb{R}$ can be expressed as follows:

$$\nabla f(\mathbf{x}) = 2[\mathrm{D}_\mathbf{x}\mathbf{r}]^{\mathrm{T}} \cdot \mathbf{r}(\mathbf{x}) \tag{2-34a}$$

$$\nabla^2 f(\mathbf{x}) = 2[\mathrm{D}_\mathbf{x}\mathbf{r}]^{\mathrm{T}} \cdot \mathrm{D}_\mathbf{x}\mathbf{r} + 2\sum_{i=1}^{m} r_i(\mathbf{x})\, \nabla^2 r_i \tag{2-34b}$$

where $\mathrm{D}_\mathbf{x}\mathbf{r} \in \mathbb{R}^{m \times n_\mathbf{x}}$ is defined as the Jacobian matrix of the scalar residuals ($r_i, i = 1 \cdots m$) with respect to the optimization variables ($x_j, j = 1 \cdots n_\mathbf{x}$). Neglecting the higher-order second term in $\nabla^2 f(\mathbf{x})$, the optimization algorithms adopted by `lsqnonlin` in MATLAB uses $2[\mathrm{D}_\mathbf{x}\mathbf{r}]^{\mathrm{T}} \cdot \mathrm{D}_\mathbf{x}\mathbf{r}$ to approximate the Hessian matrix.

From certain starting points, the `lsqnonlin` solver can find a local minimum of the objective function through the Levenberg-Marquardt algorithm, which is a combination of the steepest descent and the Gauss-Newton algorithm [54]. At every iteration, the algorithm first linearizes the objective function (Eq. (2-34)) with respective to the corresponding optimization variables. When the current solution is far from a local optimum, the Levenberg-Marquardt algorithm approaches the steepest descent algorithm. On the other hand, when the current solution is close to a local optimum, the Levenberg-Marquardt algorithm approaches the Gauss-Newton algorithm. The Levenberg-Marquardt algorithm can be used to solve the optimization problems of modal property difference and modal dynamic residual formulations. The drawback of the Levenberg-Marquardt

implementation in MATLAB is that it does not allow setting the upper and lower bounds of the optimization variables.

In addition to the Levenberg-Marquardt algorithm, `lsqnonlin` solver also provides the trust-region-reflective algorithm to solve an optimization problem [55]. The trust-region-reflective algorithm approximates the original problem with a quadratic subproblem within a small region around the current solution point, i.e. a trusted region. The quadratic subproblem is formulated using the same gradient and approximated Hessian of the original problem. By solving the quadratic subproblem using the two-dimensional subspace approach, a solution of current subproblem can be obtained [56, 57]. If the decrease of the objective function evaluated at current step is within the prescribed upper and lower bounds, the solution will be accepted, and the algorithm will continue with the next iteration. Otherwise, the trusted region at the current iteration will be adjusted, and the quadratic subproblem is solved again with the new region. Iteratively, the optimization converges to a local minimum of the objective function. The advantage of the trust-region-reflective implementation in MATLAB is that it allows users to define the upper and lower bounds of the optimization variables. However, the trust-region-reflective algorithm implemented in MATLAB cannot solve underdetermined problems. For the algorithm to work, the length of residual vector, *m*, should be at least as large as the number of variables $n_{\mathbf{x}}$ (Eq. (2-33)).

From Eqs. (2-6), (2-8) and (2-10), the optimization problem of modal property difference and modal dynamic residual formulations can be equivalently rewritten to satisfy the least-squares format required for the `lsqnonlin` solver. When using MAC value formulation in Eq. (2-6), the optimization variable **x** is the updating vector variable

$\boldsymbol{\alpha}$. So that the MAC value formulation in Eq. (2-6a) is rewritten in least squares form as $f(\boldsymbol{\alpha}) = \|\mathbf{r}(\boldsymbol{\alpha})\|_2^2$, the residual vector shown in Eq. (2-33), $\mathbf{r}(\boldsymbol{\alpha}): \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}^{2 \cdot n_{\text{modes}}}$, is formulated as a function of variable $\boldsymbol{\alpha}$. The length of the residual vector is $m = 2 \cdot n_{\text{modes}}$. The formulation of $\mathbf{r}(\boldsymbol{\alpha})$, which is previously given in Eq. (2-11), is repeated as follows in an explicit form.

$$\mathbf{r}(\boldsymbol{\alpha}) = \begin{bmatrix} w_{\lambda_1} \cdot \left(\lambda_1^{\text{EXP}} - \lambda_1(\boldsymbol{\alpha})\right) / \lambda_1^{\text{EXP}} \\ w_{\boldsymbol{\psi}_1} \cdot \left(1 - \sqrt{\text{MAC}_1(\boldsymbol{\alpha})}\right) / \sqrt{\text{MAC}_1(\boldsymbol{\alpha})} \\ \vdots \\ w_{\lambda_{n_{\text{modes}}}} \cdot \left(\lambda_{n_{\text{modes}}}^{\text{EXP}} - \lambda_{n_{\text{modes}}}(\boldsymbol{\alpha})\right) / \lambda_{n_{\text{modes}}}^{\text{EXP}} \\ w_{\boldsymbol{\psi}_{n_{\text{modes}}}} \cdot \left(1 - \sqrt{\text{MAC}_{n_{\text{modes}}}(\boldsymbol{\alpha})}\right) \Big/ \sqrt{\text{MAC}_{n_{\text{modes}}}(\boldsymbol{\alpha})} \end{bmatrix} \tag{2-35}$$

When using the eigenvector difference formulation in Eq. (2-8), the optimization variable $\mathbf{x}$ is the updating vector variable $\boldsymbol{\alpha}$. For the eigenvector difference formulation in Eq. (2-8a) to be rewritten in the form as $f(\boldsymbol{\alpha}) = \|\mathbf{r}(\boldsymbol{\alpha})\|_2^2$, the residual vector shown in Eq. (2-33), $\mathbf{r}(\boldsymbol{\alpha}): \mathbb{R}^{n_{\boldsymbol{\alpha}}} \to \mathbb{R}^{n_m \cdot n_{\text{modes}}}$, is formulated as a function of $\boldsymbol{\alpha}$. The formulation is previously given in Eq. (2-25) and repeated as follows.

$$\mathbf{r}(\boldsymbol{\alpha}) = \begin{bmatrix} w_{\lambda_1} \cdot \left(\lambda_1^{\text{EXP}} - \lambda_1(\boldsymbol{\alpha})\right) / \lambda_1^{\text{EXP}} \\ w_{\boldsymbol{\psi}_1} \cdot \mathbf{Q}_1 \left\{\boldsymbol{\psi}_1^{\text{EXP,m}} - \boldsymbol{\psi}_1^{\text{m}}(\boldsymbol{\alpha})\right\} \\ \vdots \\ w_{\lambda_{n_{\text{modes}}}} \cdot \left(\lambda_{n_{\text{modes}}}^{\text{EXP}} - \lambda_{n_{\text{modes}}}(\boldsymbol{\alpha})\right) / \lambda_{n_{\text{modes}}}^{\text{EXP}} \\ w_{\boldsymbol{\psi}_{n_{\text{modes}}}} \cdot \mathbf{Q}_{n_{\text{modes}}} \left\{\boldsymbol{\psi}_{n_{\text{modes}}}^{\text{EXP,m}} - \boldsymbol{\psi}_{n_{\text{modes}}}^{\text{m}}(\boldsymbol{\alpha})\right\} \end{bmatrix} \tag{2-36}$$

Here each term $w_{\boldsymbol{\psi}_i} \cdot \mathbf{Q}_i\{\boldsymbol{\psi}_i^{\mathrm{EXP,m}} - \boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\}$ is a $(n_m - 1) \times 1$ vector. As a result, the length of the residual vector is $m = n_m \cdot n_{\mathrm{modes}}$ (recall that n_m is the number of measured/instrumented DOFs).

As for the modal dynamic residual formulation in Eq. (2-10), the optimization variable $\mathbf{x}$ includes the updating vector variable $\boldsymbol{\alpha}$ and the unmeasured DOFs of the simulated eigenvectors $\boldsymbol{\varphi}^{\mathrm{u}}$. To rewrite the modal dynamic residual formulation in Eq. (2-10a) in the form as $f(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) = \|\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}})\|_2^2$, the residual vector shown in Eq. (2-33), $\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}): \mathbb{R}^{n_\alpha + n_{\mathrm{u}} \cdot n_{\mathrm{modes}}} \rightarrow \mathbb{R}^{N \cdot n_{\mathrm{modes}}}$, is formulated as a function of $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$. The formulation is previously given in Eq. (2-28) and repeated explicitly as follows, where the length of the residual vector is $m = N \cdot n_{\mathrm{modes}}$.

$$\mathbf{r}(\boldsymbol{\alpha}, \boldsymbol{\varphi}^{\mathrm{u}}) = \begin{bmatrix} w_1 \cdot [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_1^{\mathrm{EXP}} \mathbf{M}] \begin{Bmatrix} \boldsymbol{\psi}_1^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_1^{\mathrm{u}} \end{Bmatrix} \\ w_2 \cdot [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_2^{\mathrm{EXP}} \mathbf{M}] \begin{Bmatrix} \boldsymbol{\psi}_2^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_2^{\mathrm{u}} \end{Bmatrix} \\ \vdots \\ w_{n_{\mathrm{modes}}} \cdot \left[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_{n_{\mathrm{modes}}}^{\mathrm{EXP}} \mathbf{M}\right] \begin{Bmatrix} \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{u}} \end{Bmatrix} \end{bmatrix} \tag{2-37}$$

Finally, at each step of the optimization process, by default, `lsqnonlin` calculates the search gradient, $\nabla f(\mathbf{x})$, of the objective function numerically using the finite difference method [58]. For the MAC value and eigenvector difference formulations, as mentioned in Section 2.2, the objective function shown in Eqs. (2-6) and (2-8) are an oracle form of updating variable $\boldsymbol{\alpha}$. Therefore, the numerically calculated gradient results are affected by the difference $\Delta\mathbf{x}$, i.e. step size of $\mathbf{x}$, and prone to inaccuracies. Meanwhile, instead of using the numerically calculated gradient, `lsqnonlin` also accepts user-

provided analytical formulation of the gradient. Given that gradient simply equals the transpose of Jacobian, i.e. $\nabla f(\mathbf{x}) = (\mathrm{D}_{\mathbf{x}} f)^{\mathrm{T}}$, the definition of $\mathrm{D}_{\boldsymbol{\alpha}} f$ in Section 2.3.1 as well as $\mathrm{D}_{\boldsymbol{\alpha}} \mathbf{r}$ in Eq. (2-14**)** can be used to calculate the analytical gradient for the MAC value formulation. Similarly, definition of $\mathrm{D}_{\boldsymbol{\alpha}} f$ in Section 2.3.2 as well as $\mathrm{D}_{\boldsymbol{\alpha}} \mathbf{r}$ in Eq. (2-25) can be used to calculate the analytical gradient for the eigenvector difference formulation. Finally, for the modal dynamic residual formulation, Eq. (2-10) is written explicitly in terms of optimization variables, $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$. In addition, by observing Jacobian shown in Eqs. (2-30) and (2-31), it can be concluded that the numerical gradient calculated by the finite difference method is theoretically the same as the analytical gradient. Nevertheless, numerically calculated gradient requires more computational power.

2.5 Numerical Studies

To investigate the performance of modal property difference and modal dynamic residual formulations for model updating, numerical studies are conducted. The first example is a lumped mass-spring structure, the second one is a steel pedestrian bridge, and the last one is a concrete building frame. Table 2-1 summarizes the applicable algorithms for the three formulations (Eqs. (2-6), (2-8) and (2-10)). When solving the optimization problems of each formulation, both the Levenberg-Marquardt and the trust-region-reflective algorithms are adopted for comparison. For example, Case 2(b) means applying the trust-region-reflective algorithm onto the eigenvector difference formulation. For both algorithms, the effect of using numerical or analytical gradient on the MAC value and eigenvector difference formulations (Case 1 and 2) will be compared on the model updating of the concrete building frame. For the modal dynamic residual formulation (Case 3), as mentioned in Section 2.4, the numerical gradient is equal to the analytical one, but requires

more computational power. Therefore, optimization algorithms will only use analytical gradient for Case 3 throughout the numerical studies. Furthermore, as mentioned in Section 2.2, objective functions in Eqs. (2-6), (2-8) and (2-10) are all nonconvex. Therefore, global optimality of the optimal solution from the local search optimization algorithms cannot be guaranteed. To increase the chance of finding the global minimum, optimization process will be initiated from many feasible and randomized starting values of the updating variables.

Table 2-1 Model updating algorithms for objective function in Section 2.2

Optimization algorithms	MAC value formulation Eq.(2-6)	Eigenvector difference formulation Eq.(2-8)	Modal dynamic residual formulation Eq. (2-10)
Levenberg-Marquardt (Cannot set bounds for optimization variables)	Case 1(a)	Case 2(a)	Case 3(a)
trust-region-reflective (Cannot solve underestimated problems)	Case 1(b)	Case 2(b)	Case 3(b)

In this chapter, modal properties of the structure with actual/correct stiffness values of $\boldsymbol{\alpha}$ are used as the "experimental" properties, i.e. λ_i^{EXP} and $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ in Eqs. (2-6), (2-8) and (2-10). For practicality, only some of the DOFs are instrumented and only modes associated with the lowest few resonance frequencies are made available for model updating. In these numerical studies, the weighting factors in Eqs. (2-6), (2-8) and (2-10) are all set as 1 for simplicity, i.e. $w_{\lambda_i} = w_{\boldsymbol{\psi}_i} = w_i = 1, i = 1 \dots n_{\mathrm{modes}}$.

2.5.1 Lumped Mass-Spring Structure

Figure 2-3 shows a 6-DOF lumped mass-spring structure. In the nominal model, all the mass and nominal spring stiffness values are set to 6kg and 35kN/m, respectively.

Figure 2-3 also shows the instrumentation locations on the structure, where 3 DOFs are assumed to be measured, i.e. the length of both $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ is $n_{\mathrm{m}} = 3$.

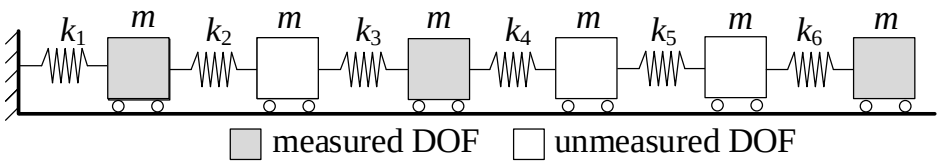


Figure 2-3. Lumped mass-spring model and sensor instrumentation

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Table 2-2 lists the stiffness parameters to be updated, which are the stiffness values of spring $k_1 \sim k_6$, and the table also shows the nominal and true/actual spring stiffness values.

Table 2-2 Structural properties of the lumped mass-spring structure

Stiffness parameters		Nominal value	Actual value	Updating variables	α_i^{act}
Spring stiffness value (kN/m)	k_1	35.00	26.25	α_1	-0.25
	k_2	35.00	28.00	α_2	-0.20
	k_3	35.00	29.75	α_3	-0.15
	k_4	35.00	31.50	α_4	-0.10
	k_5	35.00	33.25	α_5	-0.05
	k_6	35.00	36.75	α_6	0.05

Corresponding to the 6 stiffness parameters being updated, the $n_{\boldsymbol{\alpha}} = 6$ number of updating variables ($\alpha_1 \sim \alpha_6$) are also listed in the table. The column α_i^{act} in Table 2-2 lists the true/actual values of each α_i, i.e. the ideal solutions to be identified from FE model updating. For example, the ideal solution of α_1 is calculated as the relative change of the

actual k_1 value from the nominal value: $\alpha_1^{\text{act}} = \frac{k_1^{\text{act}} - k_1^{\text{nom}}}{k_1^{\text{nom}}} = \frac{26{,}250 - 35{,}000}{35{,}000} = -0.25$, i.e. a 25% reduction from the nominal value.

It is assumed that only the first three vibration modes ($n_{\text{modes}} = 3$) are available for model updating. As shown in Section 2.4, the Levenberg-Marquardt algorithm does not have difficulty with underestimated problems ($m < n_{\mathbf{x}}$ in Eq. (2-33)). The algorithm is always applicable to modal property difference and modal dynamic residual formulations, i.e. Case 1(a), 2(a) and 3(a) in Table 2-1 are all applicable. On the other hand, the trust-region-reflective algorithm is applicable only if the length of residual vector is no less than the number of optimization variables, i.e. $m \geq n_{\mathbf{x}}$ in Eq. (2-33). First take applying the trust-region-reflective algorithm on MAC formulation as an example, denoted as Case 1(b) in Table 2-1. Eq. (2-35) shows that the length of the residual vector, *m* equals $2 \cdot n_{\text{modes}} = 6$, while the number of optimization variables ($n_{\mathbf{x}}$) equals $n_{\boldsymbol{\alpha}} = 6$, and thus, $n_{\mathbf{x}} = m$. For the eigenvector difference formulation (Case 2(b) in Table 2-1), Eq. (2-36) presents that the length of the residual vector m equals $n_m \cdot n_{\text{modes}} = 9$. The number of optimization variable $n_{\mathbf{x}}$ still equals $n_{\boldsymbol{\alpha}} = 6$, making the optimization problem not underdetermined. Finally, for the modal dynamic residual approach, Case 3(b) in Table 2-1, as shown in Eq. (2-37), the number of optimization variable ($n_{\mathbf{x}}$) increases to $n_{\boldsymbol{\alpha}} + n_{\text{u}} \cdot n_{\text{modes}} = 15$. Nevertheless, with the length of the residual vector, m, also increases to $N \cdot n_{\text{modes}} = 18$, m is still larger than $n_{\mathbf{x}}$. Therefore, all Case 1(b), 2(b) and 3(b) are applicable for this lumped mass-spring structure.

When using MATLAB `lsqnonlin` with the trust-region-reflective algorithm, the upper and lower bounds of $\boldsymbol{\alpha}$ are simply set to be 1 and -1, respectively. This means that

the actual stiffness parameters are assumed to be within ±100% of the nominal values. For the modal dynamic residual formulation, the upper and lower bounds of $\boldsymbol{\varphi}^{\mathrm{u}}$ are set to be 2 and -2. We assume the experimental eigenvector, $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$, is first normalized so that the largest magnitude is 1, and that at least one DOF with large amplitude in this eigenvector is instrumented. For each applicable case in Table 2-3, the optimization process is initiated from 100 random starting points of $\boldsymbol{\alpha}$, which are uniformly randomly generated between the upper and lower bounds. For the modal dynamic residual formulation, the initial value of $\boldsymbol{\varphi}^{\mathrm{u}}$ is obtained by solving Eq. (2-5) with the random initial value of $\boldsymbol{\alpha}$. On the other hand, when using the Levenberg-Marquardt algorithm with MATLAB `lsqnonlin`, upper and lower bounds cannot be handled by the toolbox, as described in Section 2.4. Consequently, optimal result sets obtained from Levenberg-Marquardt algorithm that are out of the bounds are discarded and not included in the final result sets. Instead, the starting point is replaced with the next randomly generated point that can conclude the search within the desired bounds of optimization variables. As a result, for Case 1(a), 2(a) and 3(a) that use Levenberg-Marquardt algorithm, in order to obtain 100 sets of optimal results that are within the bounds, the Levenberg-Marquardt algorithm may have to solve the optimization problems from more than 100 starting points.

2.5.1.1 Updating Results of MAC Value Formulation

As shown in Table 2-1, Case 1(a) and 1(b) represent solving the MAC value formulation with Levenberg-Marquardt and trust-region-reflective algorithms, respectively. Using α_i^* to represent the optimal solution of each search, the relative error of every stiffness parameter can be calculated as the relative difference of the updated

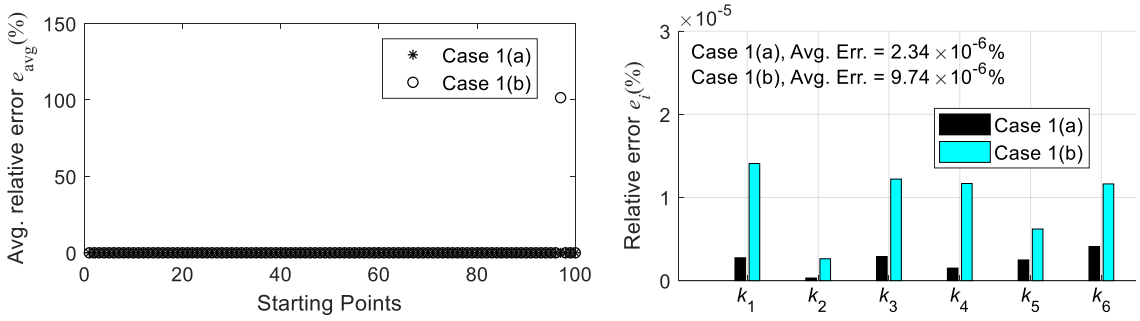
stiffness parameter value from the actual stiffness parameter value. For example, if $k_1^* = k_1^{\text{nom}} \cdot (1 + \alpha_i^*)$ is the optimal stiffness value of k_1, the corresponding relative error is calculated based on the actual/correct stiffness value, $k_1^{\text{act}} = k_1^{\text{nom}} \cdot (1 + \alpha_i^{\text{act}})$. As a result, the relative error $e_1 = \frac{|k_1^* - k_1^{\text{act}}|}{k_1^{\text{act}}} \times 100\% = \frac{|\alpha_1^* - \alpha_1^{\text{act}}|}{1 + \alpha_1^{\text{act}}} \times 100\%$. In general, the relative error of the *i*-th stiffness parameter is calculated as:

$$e_i = \frac{|\alpha_i^* - \alpha_i^{\text{act}}|}{1 + \alpha_i^{\text{act}}} \times 100\%, \qquad i = 1 \cdots n_{\boldsymbol{\alpha}} \tag{2-38}$$

where α_i^{act} is the actual value of updating variable α_i, i.e. the value listed in the last column of Table 2-2.

Figure 2-4 plots the model updating results for MAC value formulation when using analytical gradient (Eq. (2-14)) during the optimization process (instead of the default numerical gradient calculated by MATLAB). Figure 2-4(a) shows the average relative error $e_{\text{avg}} = \frac{1}{n_{\boldsymbol{\alpha}}} \sum_{i=1}^{n_{\boldsymbol{\alpha}}} e_i$ among all $n_{\boldsymbol{\alpha}} = 6$ optimal stiffness parameters, for each of the 100 in-bound optimal result sets. Horizontal axis is the sequence number of 100 randomized starting points. Recall that Case 1(a) uses the Levenberg-Marquardt algorithm, for which the implementation in MATLAB does not allow setting the bounds of the optimization variables. For this problem, 5 sets of the optimal results from Case 1(a) are out of the [-1, 1] bounds, which demonstrates the nonconvexity of the MAC value formulation. The figure shows that, after discarding 5 out-of-bound result sets, all the 100 in-bound optimal result sets from Case 1(a) are almost identical to the actual values. For Case 1(b), only 1 starting point fails to converge around the actual values of stiffness parameters ($e_{\text{avg}} =$

101.5%), while other 99 starting points converge correctly to the actual stiffness values. After looking into the incorrect optimal result set, it is found that the optimization search process converges at the assigned bounds of updating variable **α**, which also implies the nonconvexity of MAC value formulation. Additionally, Figure 2-4(b) plots the relative error of each stiffness parameter (e_i) for the best solution of each case. The best solution is chosen as the set with the minimum objective function value. The relative error shown in Figure 2-4(b) is negligible for both the Levenberg-Marquardt and the trust-region-reflective algorithms, given that the average relative error equal to 2.34×10^{-6}% for Case 1(a) and 9.79×10^{-6}% for Case 1(b). On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 40 seconds to obtain 100 optimal results that are within the bounds for Case 1(a) and 14 minutes for Case 1(b).



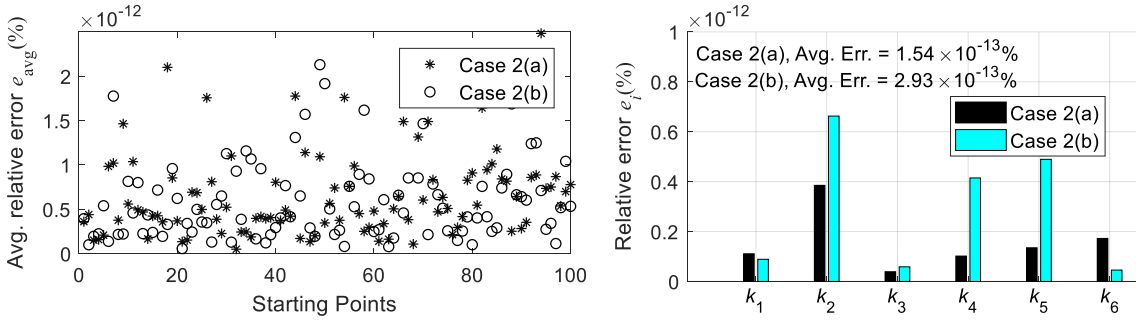
(a) Average of relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-4. Lumped mass-spring model: updating results of MAC value formulation using analytical gradient

2.5.1.2 Updating Results of Eigenvector Difference Formulation

Recall that Case 2(a) and 2(b) stand for applying Levenberg-Marquardt and trust-region-reflective algorithms on the eigenvector difference formulation, as shown in Table 2-1. Analytical gradient shown in Eq. (2-25) is adopted during each optimization process.

For each case, Figure 2-5(a) shows average relative error (e_{avg}) of the optimal result sets from 100 randomized starting points that finalized the search within the [-1, 1] bound. When applied with the Levenberg-Marquardt algorithm, ten optimal results are out of the assigned bound, and thus discarded. The out-of-bound result sets indicate the nonconvexity of the eigenvalue difference formulation. Figure 2-5(a) demonstrates that all the 100 in-bound optimal result sets from both the Levenberg-Marquardt and the trust-region reflective algorithm are almost identical to the actual values by observing that the maximum average relative error is smaller than 3×10^{-12}%. Similar to the MAC value formulation, Figure 2-5(b) plots the relative error of each stiffness parameter (e_i) for the best result set. The figure demonstrates that not only the best result sets from both optimization algorithms are close to the actual values, but also no significant difference exists between the two sets of best result. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 14 seconds to obtain 100 optimal results that are within the bounds for Case 1(a) and 16 minutes for Case 1(b).

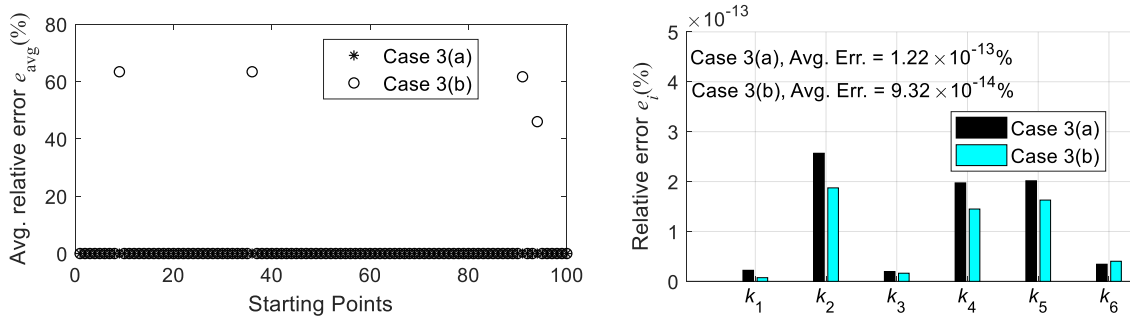


(a) Average of relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-5. Lumped mass-spring model: updating results of eigenvector difference formulation using analytical gradient

2.5.1.3 Updating Results of Modal Dynamic Residual Formulation

For this lumped spring-mass structure, both Levenberg-Marquardt and trust-region-reflective algorithms can be applied to the modal dynamic residual formulation (Case 3(a) and 3(b) in Table 2-1). The average relative error (e_{avg}) of 100 optimal stiffness parameters are plotted in Figure 2-6(a), where analytical gradient in Eqs. (2-30) and (2-32) are used during the optimization process. During the local search process, 50 optimal result sets from the Levenberg-Marquardt algorithm converge out of the bounds, i.e. [-1, 1] for $\boldsymbol{\alpha}$ and [-2, 2] for $\boldsymbol{\varphi}^{\text{u}}$, while all the 100 in-bound optimal result sets are close to the actual values. On the other hand, the trust-region-reflective algorithm can guarantee that all the optimal result sets are within the assigned bound, but 4 optimization processes fail to converge around the actual values, with e_{avg} ranging from 46% to 63%. After investigating the 4 incorrect optimal result sets, it is discovered that at least one of the optimal values for the optimization variables hits the assigned bounds. The incorrect optimal result sets from the trust-region-reflective algorithm also demonstrate the nonconvexity of the modal dynamic residual formulation. Again, for the best result set, Figure 2-6(a) plots the relative error of each stiffness parameter for both Case 3(a) and 3(b), where no obvious difference is



(a) Average of relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-6. Lumped mass-spring model: updating results of modal dynamic residual formulation using analytical gradient

observed between two result sets. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 187 seconds to obtain 100 optimal results that are within the bounds for Case 3(a) and 609 second for Case 3(b).

In conclusion, the model updating results shown in this subsection demonstrate that all the three model updating formulations can successfully update the stiffness parameters of this lumped mass-spring structure. For this numerical example, both Levenberg-Marquardt and trust-region-reflective algorithms can be applied to all three model updating formulations, i.e. all the cases in Table 2-1 are applicable. When using the Levenberg-Marquardt algorithm, for all three formulations, although some optimal result sets are out of the bounds, all the in-bound result sets are almost identical to the actual values. The out-of-bound result sets imply the nonconvexity of the three model updating formulations. Comparing among the three formulations, no obvious difference can be observed between the best optimal result sets. The computational time of Case 3(a) is longer than Case 1(a) and 2(a). The reason is that more out-of-bound result sets are rejected, i.e. 5 for Case 1(a), 10 for Case 2(a) and 50 for Case 3(a).

When using the trust-region-reflective algorithm on the MAC value and modal dynamic residual formulation, i.e. Case 1(b) and 3(b), some optimization processes converge at the incorrect values, which implies the nonconvexity of these two formulations. Again, for the best optimal result set, there is no obvious difference among the three formulations. The computational time of the modal dynamic residual formulation (Case 3(b)) is shorter than the MAC value formulation (Case 1(b)) and the eigenvector difference formulation (Case 2(b)). Finally, when we compare the performance of the Levenberg-Marquardt and the trust-region-reflective algorithms on three model updating

formulations, the relative error of the stiffness parameters for the best optimal result set is almost the same. However, the computational time of using the Levenberg-Marquardt algorithm is always shorter than using the trust-region-reflective algorithm for each case in Table 2-1.

2.5.2 Steel Pedestrian Bridge

Figure 2-7 shows the FE model of a steel pedestrian bridge, which is based on a pedestrian bridge on Georgia Tech campus. Constructed in SAP2000, a commercial structural software package, the bridge model contains 46 nodes. For both two left-end nodes, the longitudinal (x-direction) DOF are constrained, while a vertical spring (k_{z1}) is allocated to represent non-ideal boundary conditions. For the front one between the two nodes (Figure 2-7), a transverse spring (k_{y1}) is also allocated. Similarly, at the right-end side, a vertical spring (k_{z2}) is allocated at both nodes, and a transverse spring (k_{y2}) is allocated at the front one between the two nodes; both nodes are free in the longitudinal direction. In total, the FE model has 274 DOFs. Although mainly a frame structure, the segmental diagonal bracings in top plane and two side planes are truss members. The modeling software SAP2000 assigns non-zero concentrated mass only to the translational DOFs. As a result, the mass matrix (**M**) is a diagonal matrix whose diagonal entries associated with rotational DOFs equal zero. As shown in Figure 2-7, it is assumed that 7 uniaxial and 7 biaxial accelerometers are instrumented for model updating. The uniaxial accelerometer measures vertical vibration of the structure, and the biaxial accelerometer measures vertical and transverse vibration. In total, 21 out of the 274 DOFs are measured, i.e. the length of $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ is $n_{\mathrm{m}} = 21$.

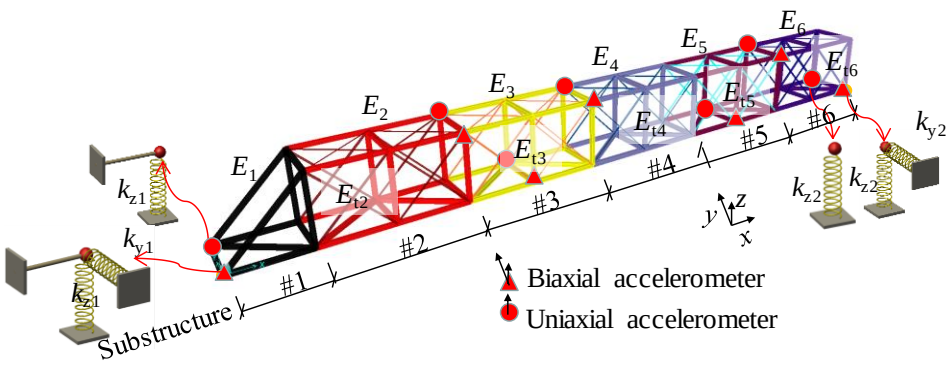


Figure 2-7. Steel pedestrian bridge model and sensor instrumentation

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Figure 2-7 shows how the entire structure is divided into six substructures for FE model updating. Substructure #1 contains only one segment from the left end of the bridge span. Other five substructures each contains two segments. Table 2-3 lists the stiffness parameters to be updated. The first substructure only contains frame members, and E_1 represents the elastic modulus of those frame members. From substructure 2 to substructure 6, each contains both frame and truss members. E_2 ~ E_6 represent the elastic moduli of frame members in each substructure, and E_{t2} ~ E_{t6} represent the elastic moduli of truss members in each substructure. The updated parameters also include the stiffness values of support springs (k_{y1}, k_{z1}, k_{y2} and k_{z2}). Table 2-3 lists the nominal and actual values of the stiffness parameters. In total, this model updating problem has 15 updating variables, i.e. $n_{\boldsymbol{\alpha}} = 15$. The column α_i^{act} in Table 2-3 lists the actual/correct values of $\boldsymbol{\alpha}$ (that are to be identified through model updating).

Table 2-3 Structural properties of the steel pedestrian bridge

Stiffness parameters		Nominal value	Actual value	Updating variables	α_i^{act}
Elastic moduli of frame members (kips/in^2)	E_1	29,000	30,450	α_1	0.05
	E_2	29,000	30,450	α_2	0.05
	E_3	29,000	27,550	α_3	-0.05
	E_4	29,000	26,100	α_4	-0.10
	E_5	29,000	31,900	α_5	0.10
	E_6	29,000	24,650	α_6	-0.15
Elastic moduli of truss members (kips/in^2)	E_{t2}	29,000	33,350	α_7	0.15
	E_{t3}	29,000	27,550	α_8	-0.05
	E_{t4}	29,000	26,100	α_9	-0.10
	E_{t5}	29,000	31,900	α_{10}	0.10
	E_{t6}	29,000	23,200	α_{11}	0.20
Support spring stiffness (kips/in)	k_{y1}	400	280	α_{12}	-0.30
	k_{z1}	500	320	α_{13}	0.60
	k_{y2}	400	280	α_{14}	-0.30
	k_{z2}	500	320	α_{15}	0.60

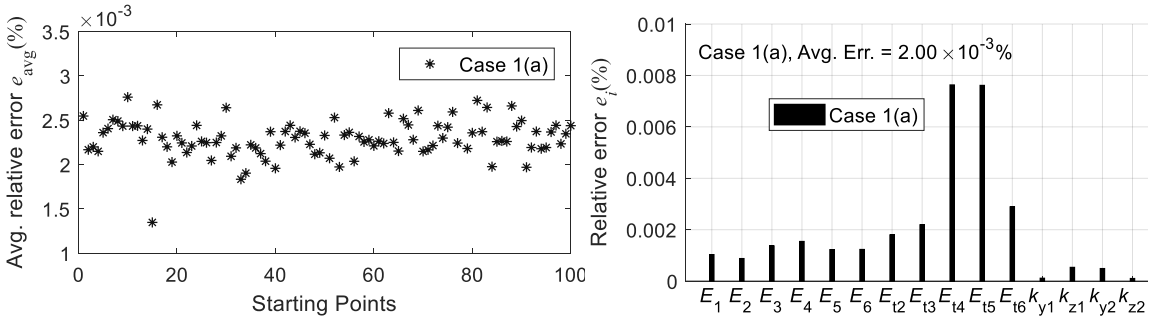
For this steel pedestrian bridge, it is assumed that the first three vibration modes ($n_{\text{modes}} = 3$) are available for model updating. As described in Section 2.4, the Levenberg-Marquardt algorithm can solve underestimated problems and thus is always applicable to Case 1(a), 2(a) and 3(a) in Table 2-1. On the other hand, the trust-region-reflective algorithm cannot solve underdetermined problems; the algorithm only works if the length of residual vector is no less than the number of optimization variables, i.e. $m \geq n_{\mathbf{x}}$ in Eq. (2-33). Consider first applying the algorithm on MAC value formulation, i.e. Case 1(b) in Table 2-3. As presented in Eq. (2-35), the length of the residual vector (m in Eq. (2-33))

equals $2 \cdot n_{\text{modes}} = 6$. Meanwhile, the number of optimization variables $n_{\mathbf{x}}$ in Eq. (2-33) equals $n_{\boldsymbol{\alpha}} = 15$. As a result, $m < n_{\mathbf{x}}$ and the problem is underdetermined. Now consider Case 2(b), applying the trust-region-reflective algorithm on the eigenvector difference formulation, as presented in Eq. (2-36), m equals $n_{\text{m}} \cdot n_{\text{modes}} = 21 \cdot 3 = 63$. The number of optimization variables $n_{\mathbf{x}}$ still equals $n_{\boldsymbol{\alpha}} = 15$. As a result, $m > n_{\mathbf{x}}$, making the problem not underdetermined. Lastly, when applying the trust-region-reflective algorithm on the modal dynamic residual formulation, as presented in Eq. (2-37), the number of optimization variables $n_{\mathbf{x}}$ increases to $n_{\boldsymbol{\alpha}} + n_{\text{u}} \cdot n_{\text{modes}} = 774$. However, with m becoming $N \cdot n_{\text{modes}} = 822$, the problem is not underdetermined. Thus, for this steel pedestrian bridge, the trust-region-reflective algorithm can be applied to Case 2(b) and 3(b), but not Case 1(b).

When using MATLAB `lsqnonlin` with the trust-region-reflective algorithm, the upper and lower bounds of $\boldsymbol{\alpha}$ are simply set to be 1 and -1, respectively. For the modal dynamic residual formulation, the upper and lower bounds of $\boldsymbol{\varphi}^{\text{u}}$ are set to be 2 and -2. In addition, for each case shown in Table 2-3, the optimization process is initiated from 100 random starting points, which are uniformly randomly generated between the upper and lower bounds of $\boldsymbol{\alpha}$. As for the modal dynamic residual formulation, the initial value of $\boldsymbol{\varphi}^{\text{u}}$ is obtained by solving Eq. (2-5) with the random initial value of $\boldsymbol{\alpha}$. Similar to the lumped mass-spring structure, when using MATLAB `lsqnonlin` with Levenberg-Marquardt, the optimal result sets that are out of the assigned bounds are rejected, and the corresponding starting point will be replaced with the next randomly generated point that can achieve in-bound optimal results.

2.5.2.1 Updating Results of MAC Value Formulation

Figure 2-8 plots the model updating results for MAC value formulation when using analytical gradient shown in Eq. (2-14). As shown in Table 2-3, Case 1(a) represents solving the MAC value formulation with Levenberg-Marquardt algorithm. Figure 2-8(a) shows the average relative error (e_{avg}) among all $n_{\boldsymbol{\alpha}} = 15$ updated stiffness parameters, for each of the 100 sets of optimal results that are within the bounds. The figure shows that, after discarding 32 sets of results that are out of [-1, 1] bound, the remaining optimal result sets are close to the correct stiffness values, i.e. $e_{\text{avg}} < 3.1 \times 10^{-3}\%$. Among the 100 sets of optimal results for each case, the best solution is chosen as the solution set with the minimum objective function value. For the best solution, Figure 2-8(b) plots the relative error of each stiffness parameter (e_i). The figure demonstrates that the best solution is almost identical to the actual stiffness value by observing that the largest and average relative error equal to $7.6\times10^{-3}\%$ and $2.0\times10^{-3}\%$, respectively. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 7 hours and 14 minutes to obtain the 100 optimal result sets that are within the bounds.

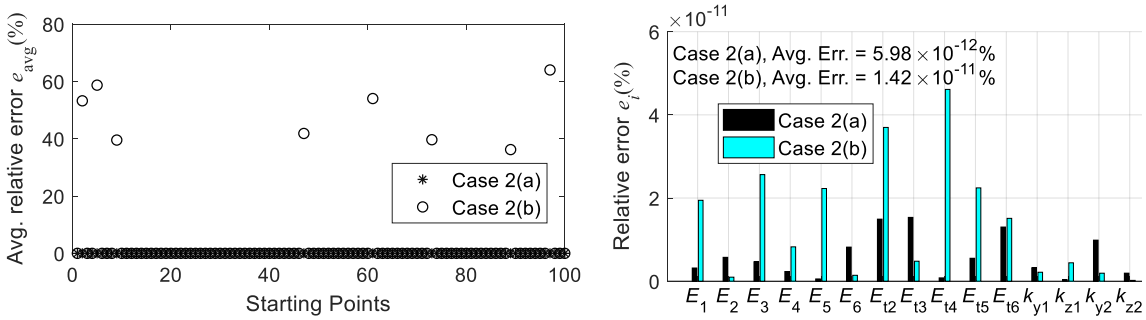


(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-8. Steel pedestrian bridge model: updating results of MAC value formulation using analytical gradient

2.5.2.2 Updating Results of Eigenvector Difference Formulation

Recall that Case 2(a) and 2(b) (Table 2-1) stand for applying Levenberg-Marquardt and trust-region-reflective algorithms, respectively, to the eigenvector difference formulation. For each optimization case, Figure 2-9(a) shows the average relative error (e_{avg}) of the optimal result sets from 100 randomized starting points. Analytical gradient shown in Eq. (2-26) is used during the optimization process. For Case 2(a) with the Levenberg-Marquardt algorithm, 61 sets of optimal results are out of the [-1, 1] bound and hence discarded. Figure 2-9(a) demonstrates that the final 100 in-bound result sets from Case 2(a) end up at the correct values of the updating variables ($e_{\text{avg}} < 3.09 \times 10^{-11}\%$). For Case 2(b) with the trust-region-reflective algorithm, although all the optimal result sets are guaranteed to be within the [-1, 1] bound, eight optimization searches fail to converge at the correct value. The average relative error of these six result sets are from 36% to 54%. After inspecting the eight incorrect result sets, it is found that each result set has at least one updating variable α_i close to bound (which also implies the nonconvexity of the eigenvector difference formulation). For each optimization case, a best solution among the 100 optimal result sets is selected by the smallest objective function value, and Figure 2-9(b) plots the relative errors of each stiffness parameter for the two best solutions. Both best solution sets provide correct stiffness values, with the average relative error equal to 5.98×10^{-12} % for Case 2(a) and 1.42×10^{-11} % for 2(b). Using the PC with an Intel i7-7700 CPU and 16 GB RAM, it takes only 17 minutes for Case 2(a) to obtain the 100 optimal solutions within the bounds and 1 hours and 21 minutes for Case 2(b). For this problem, using Levenberg-Marquardt algorithm takes much shorter computational time, while providing accurate results.



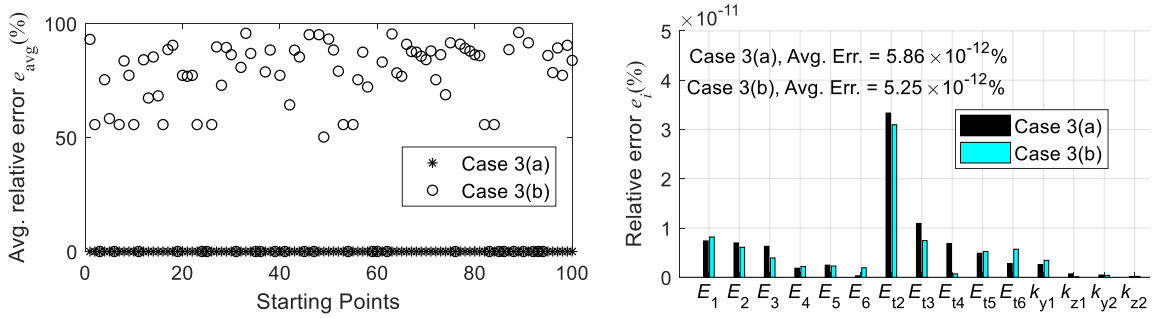
(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-9. Steel pedestrian bridge model: updating results of eigenvector difference formulation using analytical gradient

2.5.2.3 Updating Results of Modal Dynamic Residual Formulation

For this steel pedestrian bridge, both the Levenberg-Marquardt and the trust-region-reflective algorithms can be applied to the modal dynamic residual formulation, denoted as Case 3(a) and 3(b) in Table 2-1. Figure 2-10 plots the model updating results of the modal dynamic residual formulation where analytical gradient in Eqs. (2-30) and (2-32) is used during each optimization search. Figure 2-10(a) first shows the average relative error (e_{avg}) of optimal stiffness parameters from 100 in-bound optimal result sets. For Case 3(a), it is found that 102 optimal result sets are out of the bound for the optimization variables, i.e. [-1, 1] for $\boldsymbol{\alpha}$ and [-2, 2] for $\boldsymbol{\varphi}^{\text{u}}$, and thus discarded. Nevertheless, Figure 2-10(a) demonstrates that all the in-bound result sets are close to the actual values listed in Table 2-3. On the other hand, 74 out of the 100 optimal result sets from Case 3(b) converge at incorrect values, i.e. $e_{\text{avg}} > 50\%$. Similar to the lumped mass-spring structure, at least one of the optimal values of optimization variables in the incorrect result sets hits its assigned bound, which manifests the nonconvexity of the modal dynamic residual formulation. Figure 2-10(b) plots the relative error of each stiffness parameter for the result set with minimum objective function value. It can be concluded from the figure that the best

solution set for both Case 3(a) and 3(b) are almost identical to the actual stiffness parameter values, with an average relative error equal to 5.86×10^{-12}% for Case 3(a) and 5.25×10^{-12}% for Case 3(b). Using the PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 16 minutes for Case 3(a) to obtain the 100 optimal solutions within the bounds and 18 minutes for Case 3(b).



(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-10. Steel pedestrian bridge model: updating results of modal dynamic residual formulation using analytical gradient

To summarize the model updating of the steel pedestrian bridge, we first discuss the application of Levenberg-Marquardt algorithm on the MAC value, eigenvector difference and modal dynamic residual formulations, i.e. Cases 1(a), 2(a) and 3(a). Although for each formulation some out-of-bound optimal result sets are rejected, all the in-bound results can converge around the correct values of updating parameters. Comparing among the three formulations, the accuracy of optimal result sets from the modal dynamic residual formulation is higher than the other two formulations, and the computational time to obtain the 100 in-bound optimal result sets is also the shortest among three formulations.

Recall that for this example the trust-region-reflective algorithm cannot be applied to the MAC value formulation, i.e. Case 1(b) is not applicable. When using the trust-region-

reflective algorithm on both the eigenvector difference formulation – Case 2(b) and the modal dynamic residual formulation – Case 3(b), some of the optimal search results converge to the bounds, which implies the nonconvexity of the objective functions. When comparing between Case 2(b) and 3(b), more optimal results from Case 2(b) can converge to the correct values, while the computational time of Case 3(b) is much shorter than Case 2(b). We also compare Cases 2(a) and 2(b), i.e. applying the two optimization algorithms on the eigenvector difference formulation. The Levenberg-Marquardt algorithm appears more efficient for this example. Finally, when applied on the modal dynamic residual formulation, the Levenberg-Marquardt algorithm (Case 3(a)) has more in-bound optimal result sets converging at the correct values, and consumes less computational time on this example.

2.5.3 Concrete Building Frame

The last structural model studied in this chapter is a concrete building frame (Figure 2-11), which simulates a full-scale test structure in the Structural Engineering and Materials Laboratory on Georgia Tech campus. The test frame structure consists of two bays and two stories, and it was meant to be representative of low-rise reinforced concrete office buildings in the central and eastern United States built in the 1950s-1970s [59]. The columns and beams are modeled with frame elements. Corresponding to dense sensor instrumentation, seven segments are allocated per column on each story, and twelve segments per beam in each bay. In SAP2000, to ensure stiffness contribution from both concrete and steel reinforcement, along every column or beam segment, one frame element is assigned for the concrete material and another frame element is assigned for the steel reinforcement. Each floor slab is meshed into 175 shell elements. In total, the FE model

of the concrete building frame has 2,302 DOFs. Similar to the FE model of the steel pedestrian bridge, the mass matrix (**M**) is a diagonal matrix whose diagonal entries associated with rotational DOFs equal zero. Figure 2-11 also shows the accelerometer instrumentation for this simulation study, and the corresponding measurement directions. A total of 43 DOFs are measured, i.e. the length of $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ is $n_{\mathrm{m}} = 43$.

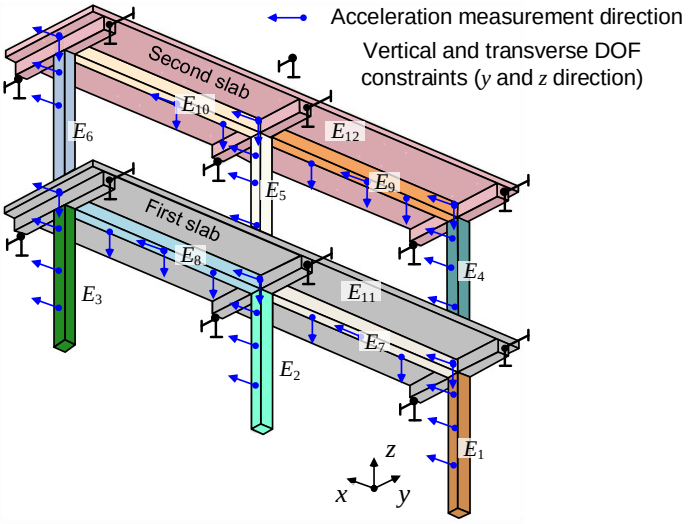


Figure 2-11. Model of a 2-story 2-bay concrete building frame [59] (height in *z*: 2 × 12 ft.; length in *x*: 2 ×18 ft.; width in *y*: 9 ft.) and sensor instrumentation

As shown in Figure 2-11, the accelerometers measure longitudinal and vertical vibration, i.e. *x* and *z* directions. Thus, only in-plane vibration mode shapes, i.e. in *x*-*z* plane, can be extracted from measurement data. To avoid the side effect of out-of-plane mode shapes (in *y*-*z* plane) on FE model updating, the vertical and transverse DOFs (*y* and *z* direction) at both ends of the three transverse beams (along *y* direction) on each slab are constrained. Lastly, at the bottom node of three columns, all six DOFs are constrained to represent an ideal fixed support.

Table 2-4 lists the stiffness parameters to be updated. As shown in Figure 2-11, in the first story, E_1 ~ E_3 represent the concrete elastic moduli of members in the three columns; E_7 and E_8 represent the concrete elastic moduli of longitudinal beam members (along *x* direction); E_{11} represents the concrete elastic moduli of the first slab and the associated transverse beam members (along *y* direction). Similarly, other moduli for the second story can be found in the figure. While this study only involves simulation, the selection of moduli corresponds to different concrete pours during the construction, and thus is in preparation for future model updating of the as-built structure with experimental data. Compared to concrete, the elastic modulus of steel reinforcement is considered to be accurate enough, and thus not being updated in this study.

Table 2-4 Structural properties of the concrete building frame

Stiffness parameters		Nominal value	Actual value	Updating variables	α_i^{act}
Elastic moduli of concrete members (kips/in^2)	E_1	3,900	3,510	α_1	-0.10
	E_2	3,700	4,440	α_2	0.20
	E_3	3,700	4,440	α_3	0.20
	E_4	3,200	3,040	α_4	-0.05
	E_5	3,200	3,840	α_5	0.20
	E_6	3,200	3,680	α_6	0.15
	E_7	3,200	3,680	α_7	0.15
	E_8	3,200	3,520	α_8	0.10
	E_9	3,400	3,060	α_9	-0.10
	E_{10}	3,400	2,890	α_{10}	-0.15
	E_{11}	3,200	3,840	α_{11}	0.20
	E_{12}	3,400	3,910	α_{12}	0.15

For all the concrete moduli being updated, Table 2-4 lists the nominal and actual values. In total, there are 12 updating variables for this model updating, i.e. $n_{\boldsymbol{\alpha}} = 12$. The column α_i^{act} in Table 2-4 lists the actual values of $\boldsymbol{\alpha}$, i.e. the ideal solutions to be identified through FE model updating.

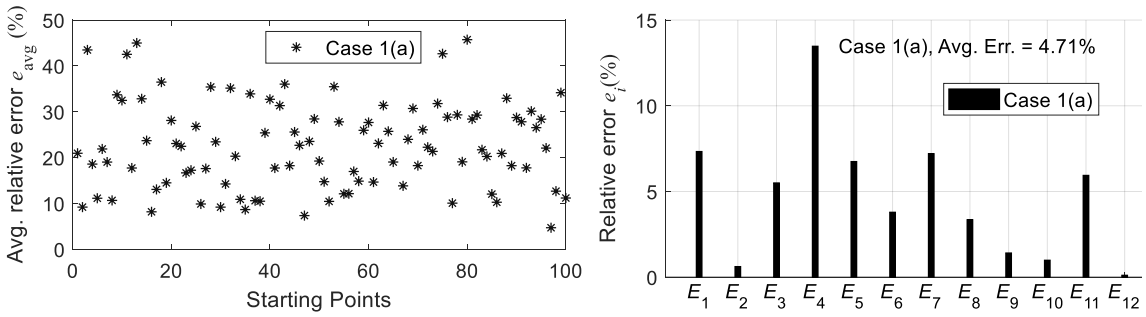
For updating the FE model, it is assumed that the first three vibration modes ($n_{\text{modes}} = 3$) are available. As shown in Eq. (2-35), when using MAC value formulation (Eq. (2-6)) to perform FE model updating, the length of the residual vector (*m* in Eq. (2-33)) equals $2 \cdot n_{\text{modes}} = 6$. Meanwhile, $n_{\mathbf{x}}$ in Eq. (2-33) equals $n_{\boldsymbol{\alpha}} = 12$. As a result, $m < n_{\mathbf{x}}$; the trust-region-reflective is not applicable for the MAC value formulation (Case 1(b) in Table 2-1). For the eigenvector difference formulation in Eq. (2-8), as presented in Eq. (2-36), the residual vector length *m* equals $n_{\text{m}} \cdot n_{\text{modes}} = 43 \cdot 3 = 129$. As a result, with the number of optimization variables $n_{\mathbf{x}}$ still equal to $n_{\boldsymbol{\alpha}} = 12$, the problem is not underdetermined; the trust-region-reflective algorithm can be applied to the eigenvector difference formulation (Case 2(b) in Table 2-1). For the modal dynamic residual formulation in Eq. (2-10), Eq. (2-37) shows that for this concrete building frame, the length of the residual vector is $m = N \cdot n_{\text{modes}} = 6{,}906$. Thus, although the number of optimization variables $n_{\mathbf{x}}$ increases from $n_{\boldsymbol{\alpha}} = 12$ to $n_{\boldsymbol{\alpha}} + n_{\text{u}} \cdot n_{\text{modes}} = 6{,}789$, the problem is not underdetermined; the trust-region-reflective algorithm can be applied to Case 3(b) in Table 2-1. Finally, the Levenberg-Marquardt algorithm can always be applied to all the cases shown in Table 2-1, i.e. Eqs. (2-6), (2-8) and (2-10).

When using MATLAB `lsqnonlin` with trust-region-reflective, the bounds of $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}^{\text{u}}$ are set to be [-1, 1] and [-2, 2], respectively. Similar to the previous two examples,

for each applicable case shown in Table 2-1, the optimization process is initiated from 100 random starting points within the bounds of $\boldsymbol{\alpha}$. The initial values of $\boldsymbol{\varphi}^{u}$ in the modal dynamic residual formulation are obtained by solving Eq. (2-5) with the random initial value of $\boldsymbol{\alpha}$. Finally, when using MATLAB `lsqnonlin` with Levenberg-Marquardt, the optimal result sets that are out of the assigned bounds are rejected, and the corresponding starting point will be replaced with the next randomly generated point that can achieve valid optimal results.

2.5.3.1 Updating Results of MAC Value Formulation

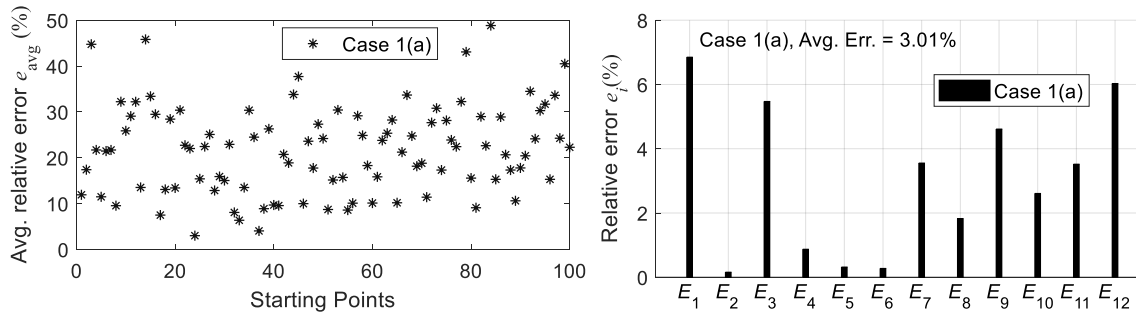
Case 1(a) is first studied, applying the Levenberg-Marquardt algorithm on the MAC value formulation. Instead of using the analytical gradient calculated by Eq. (2-14), the optimization search is first performed using gradient calculated numerically by MATLAB through finite difference method. For each of the 100 successful runs, Figure 2-12(a) displays the average relative error (e_{avg}) for all $n_{\boldsymbol{\alpha}} = 12$ stiffness parameters after discarding ten optimal result sets that are out of the [-1, 1] bounds during the optimization process. With average relative error $e_{\text{avg}} \geq 4.71\%$, the figure shows that none of the optimization processes converge close to the correct value of updating variables. For the solution set that achieves the minimum objective function value among the 100 starting points (point #97), Figure 2-12(b) plots the relative error (e_i) of each optimal stiffness parameter. The figure confirms the obtained stiffness parameter values are not reasonable, with the maximum relative error larger than 13%. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 18 hours and 58 minutes to obtain the 100 optimal result sets that are within the bounds.



(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-12. Concrete building frame model: updating results of MAC value formulation using numerical gradient

The Levenberg-Marquardt optimization for the MAC value formulation is then repeated with analytical gradient calculated by Eq. (2-14) instead. During the optimization process, five result sets from the Levenberg-Marquardt algorithm are out of the [-1, 1] bounds, and thus discarded. Figure 2-13(a) plots the e_{avg} for each of the 100 in-bound optimal result sets. Since all e_{avg} is no less than 3.01%, it can be concluded that again none of the optimal result sets converge close to the correct values. For the result set from starting point #24 with the smallest objective function value, Figure 2-13(b) shows the relative errors of the stiffness parameters (e_i). Although the results of using analytical gradient is better than using numerical gradient, the results are still not reasonable given



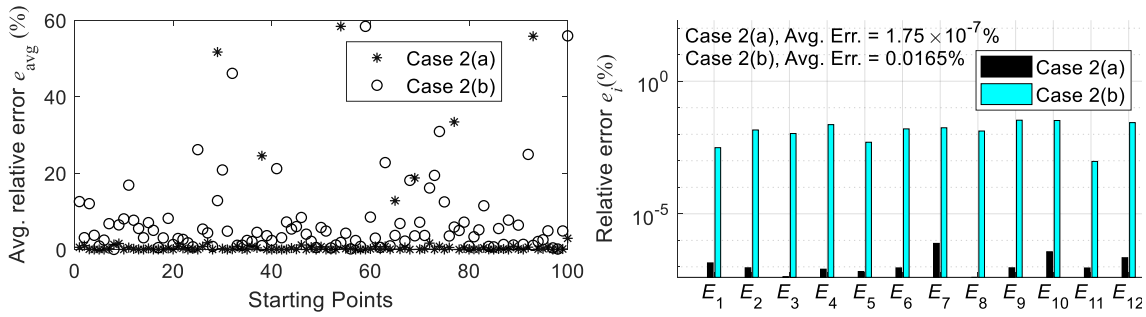
(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-13. Concrete building frame model: updating results of MAC value formulation using analytical gradient

that the maximum error is larger than 6.5%. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 22 hours and 58 minutes to obtain the 100 optimal result sets that are within the bounds.

2.5.3.2 Updating Results of Eigenvector Difference Formulation

Recall that Case 2(a) and 2(b) refer to applying Levenberg-Marquardt and trust-region-reflective algorithms, respectively, on the eigenvector difference formulation. For each of the two cases, Figure 2-14(a) plots the average relative error of $n_{\boldsymbol{\alpha}} = 12$ stiffness parameters for 100 in-bound optimal result sets. The figure is obtained when numerical gradient is used during the optimization process. For Case 2(a), after discarding nine optimal result sets that are out of the [-1, 1] bounds, 86 out of the 100 inbound result sets can update the stiffness parameter with an acceptable accuracy, i.e. $e_{\text{avg}} < 1\%$. On the other hand, for Case 2(b), only 16 optimal result sets converge around the actual values with $e_{\text{avg}} < 1\%$. For each case, the best solution is again selected as the one with the minimum objective function value among 100 result sets. The relative errors of optimal stiffness parameter values are plotted in Figure 2-14(b). It can be seen that the accuracy of optimal result sets from Case 2(a) is much higher than Case 2(b), with e_{avg} equal to

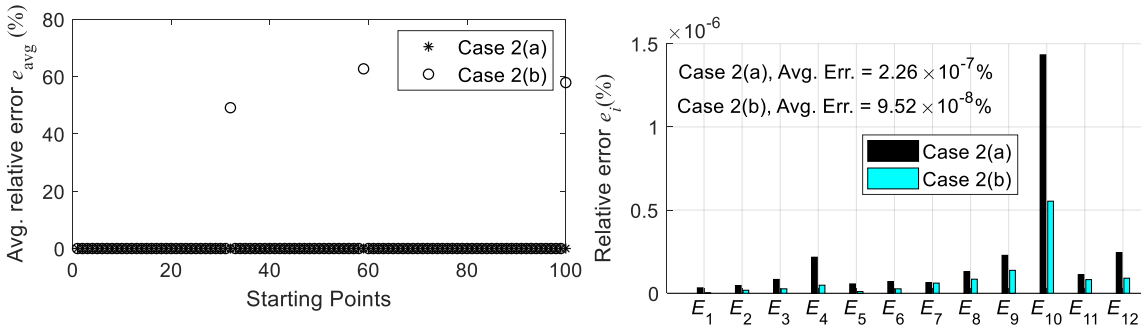


(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-14. Concrete building frame model: updating results of eigenvector difference formulation using numerical gradient

1.75×10^{-7}% and 0.0165% for the two cases, respectively. Using the same PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 23 hours and 16 minutes for Case 2(a) to obtain the 100 optimal solutions within the bounds. On the other hand, Case 2(b) took 22 hours 38 minutes.

For comparison, both optimization algorithms are repeated on the eigenvector difference formulation, but using analytical gradient in both optimization Case 2(a) and 2(b). Figure 2-15(a) plots the average relative error of all stiffness parameters for 100 optimal result sets within the [-1, 1] bounds. After discarding 14 optimal result sets that are out of the [-1, 1] bounds, all the optimal result sets from Case 2(a) converge close to the actual value with an average relative error smaller than 2.16×10^{-6}%. As for Case 2(b), the figure shows that three out of 100 optimal result sets end up with large relative error, ranging from 49.14% to 62.70%. Upon inspection of the three result sets with large errors, it is found that similar to the steel pedestrian bridge, each result set has at least one updating variable α_i hitting bound. For each optimization case, the best result with the minimum objective function value among 100 result sets is identified; the relative stiffness errors of



(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

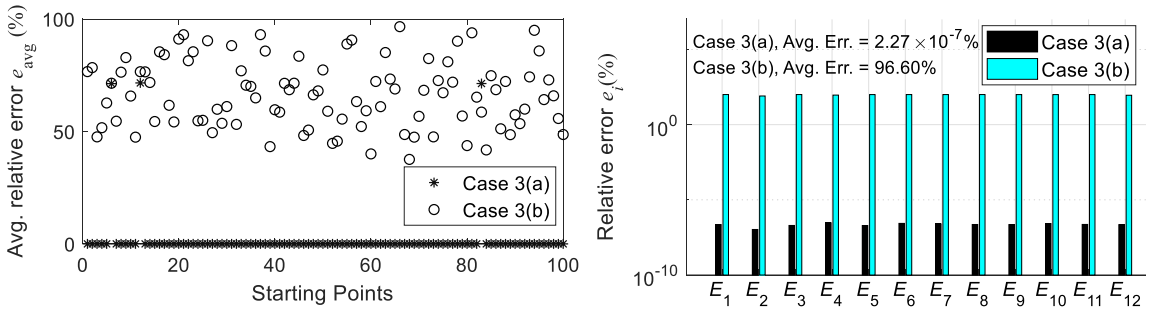
Figure 2-15. Concrete building frame model: updating results of eigenvector difference formulation using analytical gradient

each best result set are shown in Figure 2-15(b). The best solution of the two optimization cases have similar accuracy. Using the same PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 1 hours and 13 minutes for Case 2(a) to obtain the 100 optimal solutions within the bounds and 52 minutes for Case 2(b).

2.5.3.3 Updating Results of Modal Dynamic Residual Formulation

For this concrete building frame, the modal dynamic residual formulation can be applied with both Levenberg-Marquardt and trust-region-reflective algorithms, i.e. Case 3(a) and 3(b) in Table 2-1. As mentioned at beginning of this section, numerical gradient of the modal dynamic residual formulation is equivalent to the analytical gradient, but requires more computational time. Therefore, analytical gradient is used for both optimization algorithms during the optimization process, i.e. for both Case 3(a) and (b). Figure 2-16(a) plots the average relative error of all stiffness parameters for 100 optimal result sets within the bounds for both Case 3(a) and 3(b). When applied with the Levenberg-Marquardt algorithm, 46 out-of-bound optimal result sets are discarded. For the 100 in-bound optimal result sets, Figure 2-16(a) shows that 3 optimal result sets deviate away from the actual values, with e_{avg} around 70%. Meanwhile, none of the optimal results from Case 3(b) is close to the actual stiffness values, i.e. $e_{\text{avg}} > 37\%$. In addition, for each of the 100 optimal result sets from Case 3(b), at least one optimal value of optimization variables converges at its assigned bound. Therefore, the modal dynamic residual formulation is nonconvex with respect to the optimization variables. For both optimization algorithms, the best solution is selected by the smallest objective function value, and Figure 2-16(b) shows the relative error of each stiffness parameter of two best solution sets. The

best solution of the Levenberg-Marquardt algorithm (Case 3(a)) is almost identical to the actual values with an average error equal to 2.27×10^{-7}%, but the trust-region-reflective algorithm (Case 3(b)) cannot provide a reasonable result, i.e. $e_{\text{avg}} = 96.60\%$. Using the same PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 53 minutes for Case 3(a) to obtain the 100 optimal solutions within the bounds and 2 hours 3 minutes for Case 3(b).



(a) Average relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 2-16. Concrete building frame model: updating results of modal dynamic residual formulation using analytical gradient

In summary, for model updating of the concrete building frame, again only the Levenberg-Marquardt algorithm can be applied to the MAC value formulation (Case 1(a)). However, the updating cannot provide a set of optimal results with acceptable accuracy, either using numerical or analytical gradient during the optimization process. As for the eigenvector difference formulation, when using numerical gradient, both Levenberg-Marquardt (Case 2(a)) and trust-region-reflective (Case 2(b)) algorithms can find the correct updating parameter values, but the relative error of best optimal result set from the Levenberg-Marquardt algorithm is much smaller than the trust-region-reflective algorithm. On the other hand, when using analytical gradient, all the in-bound optimal result sets from Levenberg-Marquardt algorithm converge around the correct values. While a few optimal result sets from the trust-region-reflective algorithm converge to the assigned bounds, all

other results converge to the correct values. Overall, it can be concluded that when applying both optimization algorithms on the eigenvector difference formulation, using analytical gradient not only provides more accurate model updating results in general, but also can find the correct updating parameter values more efficiently. The study demonstrates the advantage of using analytical gradient versus numerical gradient. For this concrete building, both Levenberg-Marquardt and trust-region-reflective algorithms can be applied to the modal dynamic residual formulation. However, only the optimal result from Levenberg-Marquardt algorithm, i.e. Case 3(a), is reasonable. When comparing between Case 2(a) and 3(a) where the Levenberg-Marquardt algorithm is applied on the eigenvector difference and modal dynamic residual formulation, respectively, the accuracy of the best solution set is similar, but the computational time of Case 3(a) is shorter than Case 2(a).

CHAPTER 3. FE MODEL UPDATING WITH EXPERIMENTAL DATA

CHAPTER 2 introduces three finite element (FE) model updating formulations, i.e. MAC value, eigenvector difference and dynamic residual formulations. The numerical simulations in CHAPTER 2 demonstrate that using simulated modal properties of structural models, each of three FE model updating formulations can correctly identify stiffness parameter values of structural models with certain complexity. In this chapter, the FE model updating formulations will be applied on data collected from physical structures in laboratory and field experiments. To collect experimental data from actual structures, a low-cost wireless sensing node, named *Martlet*, is developed. In addition, in order to reduce the sensor cost and while maintaining accuracy, an integrated accelerometer board is developed to work together with *Martlet*.

This chapter will first introduce the hardware design of *Martlet* wireless sensing node and the integrated accelerometer board. To validate the performance of the wireless sensing node, experiments are first conducted on a laboratory four-story shear-frame structure. Experimental modal properties are then extracted from the acceleration data, which is later fed into the FE model updating formulations to update the inter-story stiffness of the structure. In addition, the wireless sensing nodes are also installed on a pedestrian bridge on Georgia Tech campus. Again, the selected stiffness parameters of the pedestrian bridge are updated with the experimental modal properties.

3.1 *Martlet* Wireless Sensing Node

In order to facilitate the safety assessment of civil structures, structural health monitoring (SHM) systems have been widely studied for monitoring structural performance and identifying potential damage [60, 61]. Among various SHM approaches, vibration-based monitoring using accelerometers plays an important role. Important structural characteristics, such as modal properties, can be extracted from the acceleration measurements [62-64].

In order to obtain more detailed structural information, it is preferred to install a large amount of sensors on the structure. Traditional SHM systems adopt lengthy coaxial cables for transmitting data from structural sensors, which results in high installation cost and is labor intensive [65]. In order to overcome the limitation of cabled SHM systems, significant efforts have been devoted to developing wireless SHM systems [1, 66, 67]. The performance of wireless SHM systems has been validated with both laboratory and field experiments [59, 68-70]. This section will introduce a recently-developed wireless sensing node, *Martlet*, together with an integrated accelerometer board for structural vibration measurement.

3.1.1 Martlet Node

Martlet, as shown in Figure 3-1, is a low-cost wireless sensing node developed for SHM applications [71]. The development of *Martlet* is a joint effort among the Laboratory for Intelligent Systems and Technologies at the University of Michigan, the Laboratory for Smart Structural Systems at Georgia Institute of Technology, and the Department of Civil and Environmental Engineering at Michigan Technological University. The *Martlet*

wireless node adopts a Texas Instruments Piccolo microcontroller as the core processor (TMS320F28069), whose clock frequency can run up to 90 MHz. The dimension of the *Martlet* node is 2.5 in × 2.25 in.

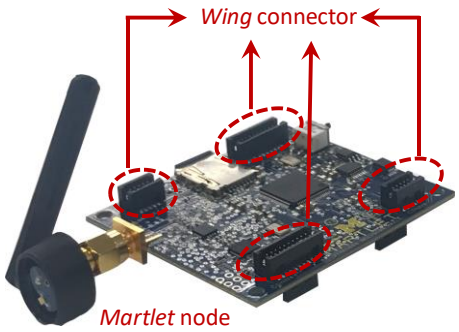


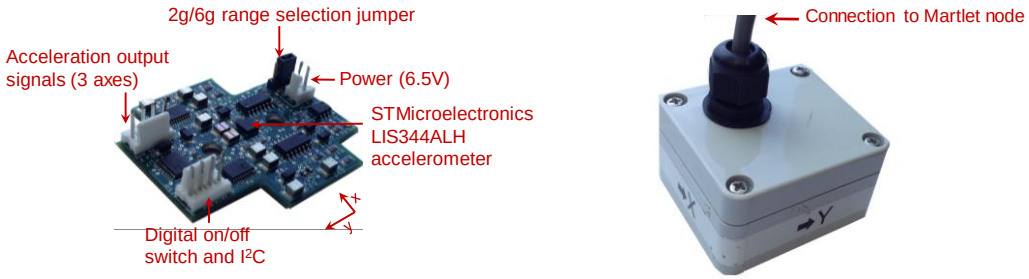
Figure 3-1. *Martlet* wireless sensor node (2.5 in × 2.25 in)

One distinct feature of the Piccolo microcontroller is the capability of high-speed data acquisition. The direct memory access (DMA) module on the microcontroller allows the *Martlet* node to collect sensor data at a sampling rate up to 3 MHz. In addition, various general purpose input/output (GPIO) pins are extended to the *wing* connectors (shown in Figure 3-1) from the microcontroller, which allow communication between the *Martlet* node and peripheral boards (termed "*wing*" boards) using protocols such as serial peripheral interface (SPI), inter-integrated circuit (I^2C), and pulse width modulation (PWM), etc. The extensible hardware design feature of the *Martlet* node enables various sensor *wing* boards to conveniently stack up through four *wing* connectors and work with the *Martlet* node. The combination of the extensible design feature with onboard 9-channel 12-bit analog-to-digital conversion (ADC) allows the *Martlet* node to simultaneously sample analog signals from multiple sensors through different sensor *wing* boards. There is 100 kB × 16-bit random access memory (RAM) available in the microcontroller for embedded computing. To extend the data storage size of the *Martlet* node, a typical micro

SD card (like these used in digital cameras) can be plugged into the *Martlet* node. The data stored in the micro SD card can be either wirelessly transferred or easily read offline by a personal computer. The *Martlet* node adopts a 2.4 GHz radio for low-power wireless communication through IEEE 802.15.4 standard [72]. The communication range can reach up to 1,600 ft at line-of-sight, and the maximum transfer rate can reach 250 kbps.

3.1.2 Integrated Accelerometer Wing

In order to obtain accurate acceleration measurement, and in the meantime reduce sensor cost, one solution is to integrate a low-cost MEMS accelerometer and specialized signal conditioning circuit into a single *wing* board, as shown in Figure 3-2(a). The integrated accelerometer *wing* adopts a tri-axial MEMS accelerometer, the STMicroelectronics LIS344ALH model. A jumper on the board selects between ±2g and ±6g measurement scales. The noise density of the measurement is 25 $\mu g/\sqrt{Hz}$ along the x-axis and y-axis, and 50 $\mu g/\sqrt{Hz}$ along the z-axis.



(a) Integrated accelerometer *wing* (2.0 in × 2.25 in)

(b) Weatherproof package (2.28in × 2.52in × 1.38in)

Figure 3-2. Integrated accelerometer *wing* with weatherproof package

The analog signals from the LIS344ALH accelerometer are directly fed into an onboard signal conditioner that performs mean shifting, low-pass filtering, and amplification (Figure 3-3). The mean shifting module is particularly useful because the

zero-g output voltage signals from the LIS344ALH accelerometer depend on the orientation of the accelerometer mount. Regardless of zero-g voltage levels of the sensor signals, the mean-shifted signals oscillate around 1.65V and the dynamic waveform remains the same as prior to shifting. Next, the anti-aliasing module prevents high-frequency signals and noises from irreversibly contaminating the digitalized data samples. A 4^{th}-order low-pass Bessel filter with a programmable cutoff frequency is adopted in this anti-aliasing design. The phase shift of a Bessel filter varies linearly with frequency. This is equivalent to a constant time delay to the signal within the passband, and thus, preserves the original waveform [73]. The cut-off frequency can be programed on-the-fly from 15Hz to a few hundred Hz. In order to improve signal-to-noise ratio, the accelerometer signal is finally amplified by a programmable amplifier. The overall amplification gain can be set from ×1.9 to ×190. A distinct feature of the integrated accelerometer *wing* is that the cutoff frequencies and gains are remotely programmable. This feature is achieved by adopting digital potentiometers (Digipots), whose resistance value can be programed on-the-fly through an I^2C interface from the *Martlet* microcontroller. The programmable cutoff frequencies and gains offer great convenience in field testing. When a new set of cutoff frequencies and gains is needed, a wireless command from the server can easily achieve immediate setting update for all *Martlet* nodes.

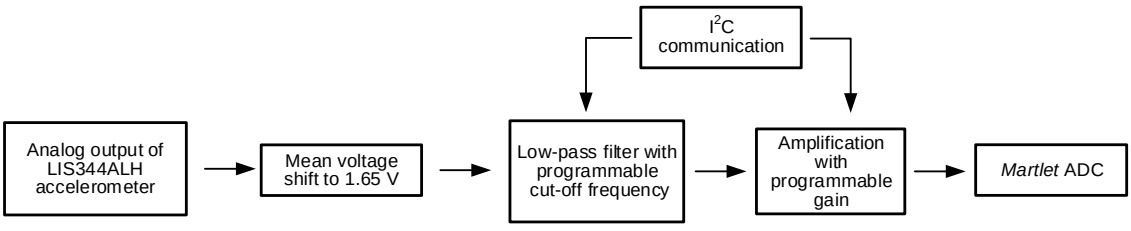
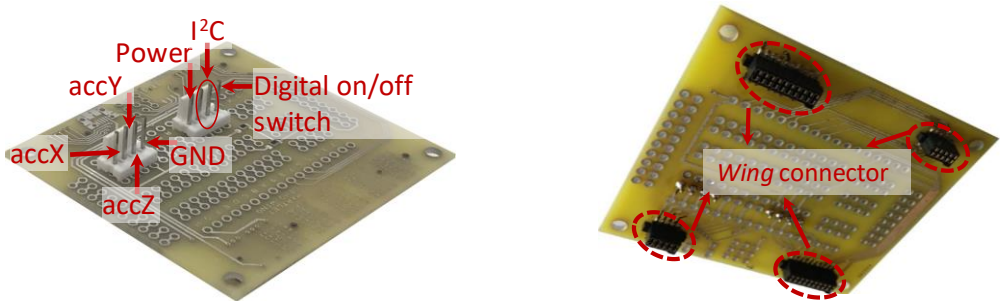


Figure 3-3. Functional diagram of the intergrated accelerometer *wing*

The integrated accelerometer *wing* is placed into a compact weatherproof enclosure with a dimension of 2.28 in (L) × 2.52 in (W) × 1.38 in (H), for firm installation of the accelerometer onto a structural surface (Figure 3-2(b)). As a result, the integrated accelerometer *wing* is connected to the *Martlet* node with an eight-wire cable. Three wires in the cable are allocated for the acceleration output signals (X, Y and Z channels), two for I^2C communication, one for power, one for ground, and the last one for a digital signal that allows the *Martlet* node to power the accelerometer *wing* on and off. An interface *wing* is developed to allow the integrated accelerometer *wing* to work with *Martlet* node (Figure 3-4). Two Molex headers are soldered on the interface *wing* for the eight-wire cable to connect to the *Martlet* node (Figure 3-4(a)). Four *wing* connectors are soldered at the bottom of the interface *wing* (Figure 3-4(b)), so that the interface *wing* can stack atop and plug onto the *Martlet* node. The current consumption of the integrated accelerometer *wing* is ~12 mA (referenced at 3.3V) under normal working conditions and ~1 μA when powered off into sleep mode.



(a) Top view (b) Bottom view

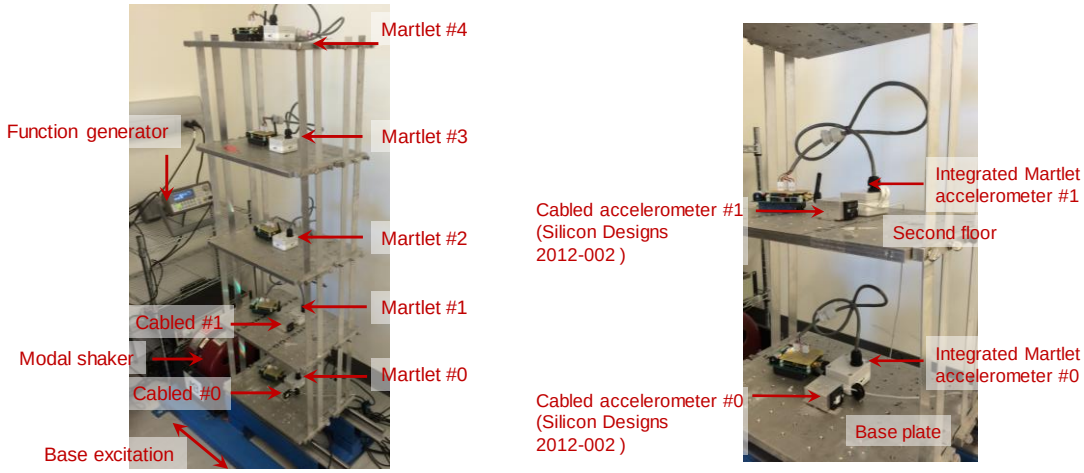
Figure 3-4. Interface *wing* between *Martlet* node and integrated accelerometer *wing* (2.5 in × 2.25 in)

3.2 FE Model Updating with Laboratory Experiment Data

In order to evaluate the performance of the *Martlet* wireless node and the integrated accelerometer *wing*, experiments are conducted on a four-story shear-frame structure in the lab. The performance of wireless sensing system is first compared with high-precision cabled system. Then, experimental modal properties are extracted from the collected structural vibration data. Finally, the inter-story stiffness of the shear-frame structure is updated with FE model updating formulations presented in CHAPTER 2 with experimentally-measured modal properties.

3.2.1 Description of Test Structure and Test Configuration

Figure 3-5 shows the four-story shear-frame structure in the lab. The structure is mounted on a shake table which generates base excitation to the structure. The entire structure is made of aluminum, including rigid plates as floors and flexible strips as columns. The properties of the structure are shown in Table 3-1.



(a) Photo of experiment setup (b) Close up view of experiment setup

Figure 3-5. Experiment setup of the four-story shear-frame structure

Table 3-1 Structural properties of the four-story shear-frame structure

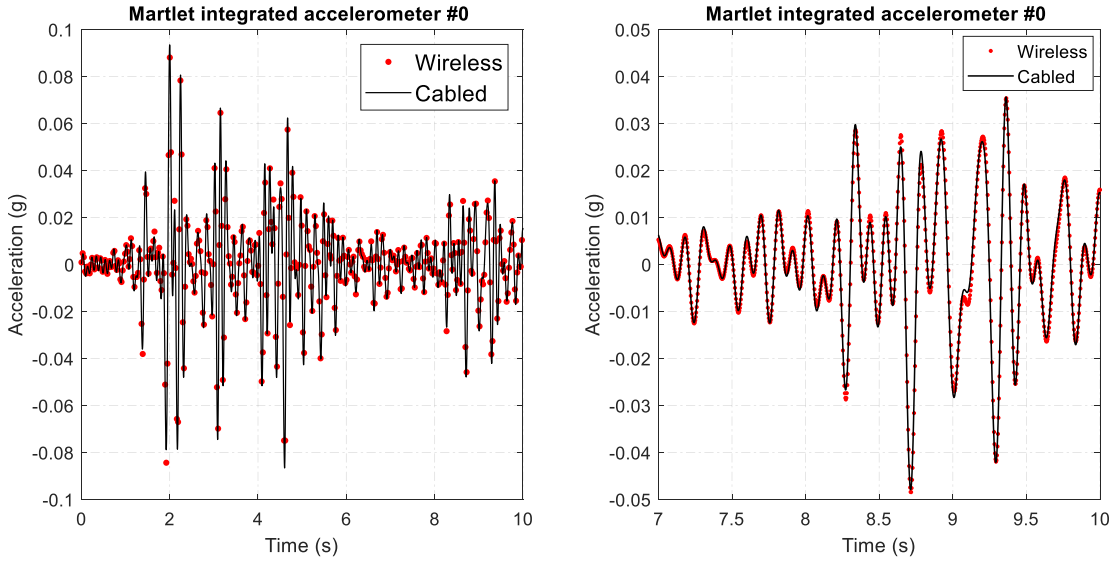
	Parameter	Value
Aluminum plate	Weight (lb)	11.310
Aluminum column	Length (in)	12.000
	Width (in)	1.000
	Thickness (in)	0.125

Figure 3-5(a) shows the sensor instrumentation on the structure. One *Martlet* node with an integrated accelerometer *wing* is installed at each floor of the structure (#1~#4 in Figure 3-5(a)). Another *Martlet* node (#0) is installed at base to measure the excitation signal generated by the modal shaker. At the base and the second floor, a high-precision cabled accelerometer (Silicon Designs 2012-002) are installed side by side with the *Martlet* node (Figure 3-5(b)). Because the modal shaker only generates single-direction ground excitation, only the x-axis of the integrated accelerometer *wing* is used to collect horizontal floor acceleration. In the following experiments, the amplification gain of the integrated accelerometer *wing*s is set to ×20, and the cutoff frequency is set as 25Hz. The sampling frequency of *Martlet* node is set as 1,000Hz. The cabled accelerometer data is sampled by a commercial National Instruments data acquisition system. A signal conditioner is connected between the accelerometer and cabled data acquisition system, the gain and cutoff frequency of which is set to be the same as the integrated accelerometer *wing*. The sampling frequency of the cabled sensing system is set as 1,652Hz.

3.2.2 Test Results and Modal Analysis

Figure 3-6 compares the time history data from the cabled accelerometers and the integrated accelerometer *wing*s. The acceleration data were collected when the modal

shaker generated the record of 1940 El Centro NS earthquake excitation to the structure. Figure 3-6(a) and (b) compare the wireless and cabled measurements on the base. Comparisons are shown for a total period of 10 seconds, and for a close-up view of 3 seconds. All figures illustrate acceptable agreement between the data sets collected by the wireless and cabled systems. It demonstrates that the integrated accelerometer *wing* with *Martlet* node is capable of providing high-quality acceleration measurements that are comparable with a high-precision cabled system.



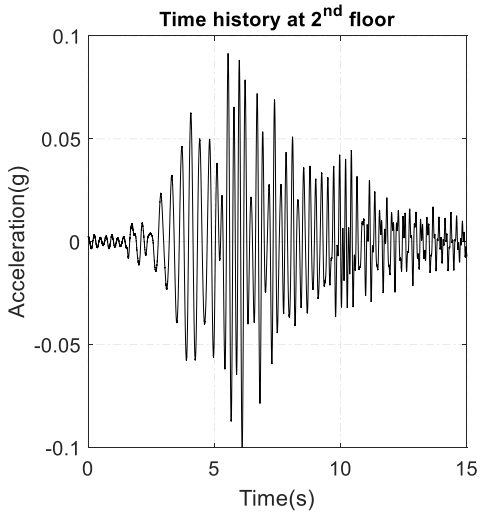
(a) Comparison for integrated accelerometer #0

(b) Close-up comparison for integrated accelerometer #0

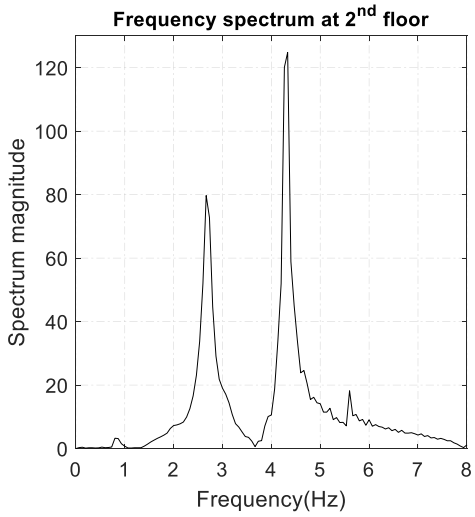
Figure 3-6. Comparison between integrated *Martlet* accelerometer *wing* and cabled accelerometer

In order to obtain acceleration data for extracting modal properties of the shear-frame structure, a chirp signal (increasing from 0Hz to 15Hz in 15s) is generated as ground excitation. During the modal test, the cabled reference accelerometers are removed from the structure; only the wireless system remains on the structure. Figure 3-7 presents two sets of example acceleration data recorded by the *Martlet* nodes installed on the 2nd and 3rd

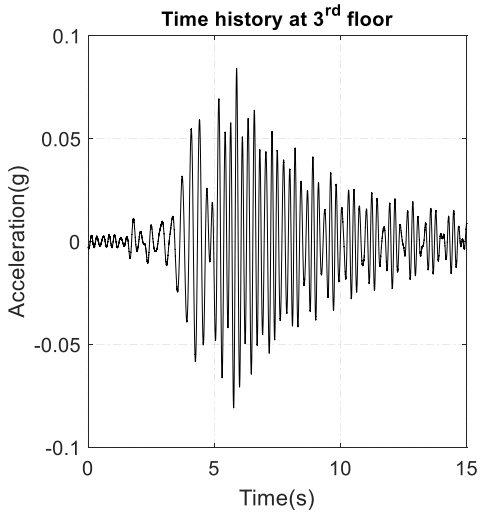
floors, as well as the corresponding frequency spectra. Similar peak resonance frequencies can be observed between the two spectra, i.e. Figure 3-7(b) and (d).



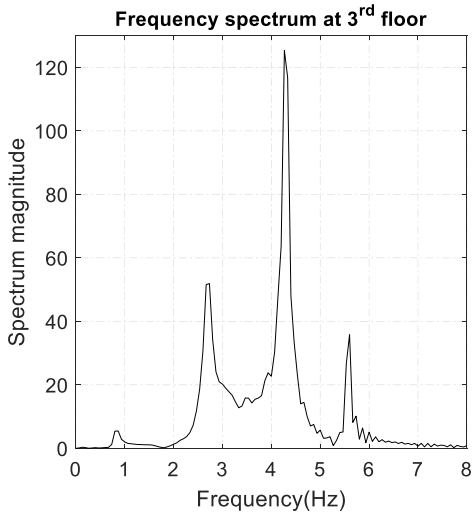
(a) Time history of 2nd floor response



(b) Frequency spectrum of 2nd floor response



(c) Time history of 3rd floor response



(d) Frequency spectrum of 3rd floor response

Figure 3-7. Example response records and corresponding frequency spectra records

The eigensystem realization algorithm (ERA) [62] is applied to the impulse response functions obtained from wireless sensing data with chirp ground excitation. Experimental modal properties of the shear-frame structure are extracted. Figure 3-8 shows the resonance frequencies and mode shapes of first four modes. The extracted resonance

frequencies match with the peaks in the example frequency spectra (Figure 3-7(b) and (d)). The mode shapes also agree with the expectation for a shear-frame structure.

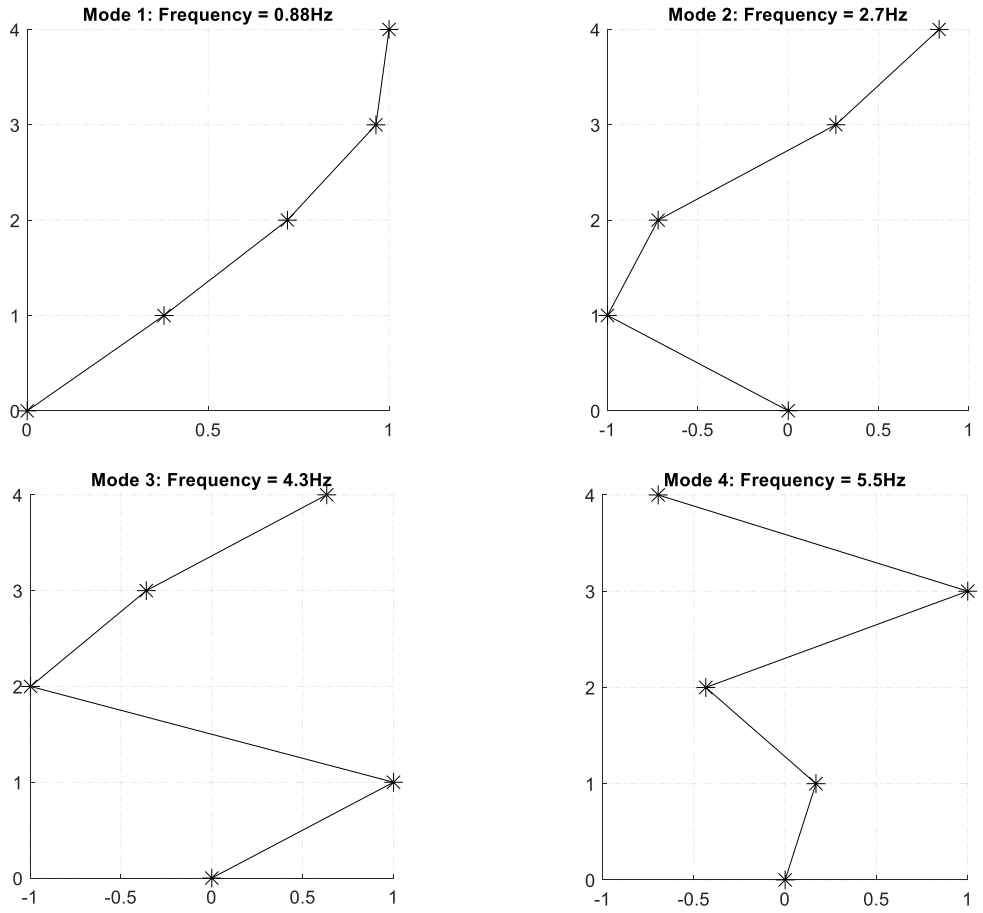


Figure 3-8. Modal properties of first four modes of the shear-frame structure

3.2.3 FE Model Updating Results

With rigid plates as floors, the four-story shear-frame structure can be simplified as a lumped mass-spring model with four degrees of freedom (DOFs), shown in Figure 3-9. In the nominal model, all the spring stiffness values are set to 10 lbf/in. The weight of each floor includes the aluminum plate and the *Martlet* sensor node with the integrated accelerometer *wing*. After being weighed by a scale, the mass of each DOF is set as 12.060

lb. Figure 3-5 shows that every floor of the structure is instrumented with sensor, so the length of both $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})$ is $n_{\mathrm{m}} = 4$ for model updating. Table 3-3 compares the experimental and simulated modal properties generated from the nominal FE model, where the MAC value is defined before in Eq. (2-7). Obvious difference can be observed in the resonance frequency of the 1st mode, and mode shapes of the 3rd and 4th modes.

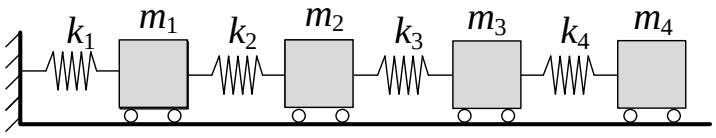


Figure 3-9. Simplified model of four-story shear-frame structure

Table 3-2 Comparison of experimental and simulated modal properties from the nominal model

Modes	Experimental results	Nominal model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.99	12.02	1.00
2nd mode	2.75	2.85	3.64	0.96
3rd mode	4.30	4.36	1.47	0.74
4th mode	5.53	5.35	3.21	0.71

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Table 3-3 lists the stiffness parameters to be updated, which are the stiffness values of spring $k_1 \sim k_4$. Corresponding to the 4 stiffness parameters being updated, the $n_{\boldsymbol{\alpha}} = 4$ number of updating variables ($\alpha_1 \sim \alpha_4$) are also listed in the table. Similar to CHAPTER 2, $\boldsymbol{\alpha}$ is scaled to represent the relative change percentage from nominal value of each stiffness parameter. All three FE model updating formulations will be adopted to update the FE model of this four-story shear-frame structure. For this laboratory structure,

the weighting factors in Eqs. (2-6), (2-8) and (2-10) are all set as 1 for simplicity, i.e. $w_{\lambda_i} = w_{\boldsymbol{\psi}_i} = w_i = 1, i = 1 \ldots n_{\text{modes}}$.

Table 3-3 Structural properties of the shear-frame structure

Stiffness parameters		Nominal value	Updating variables
Spring stiffness value (lbf/in)	k_1	10.00	α_1
	k_2	10.00	α_2
	k_3	10.00	α_3
	k_4	10.00	α_4

It is assumed that all four vibration modes shown in Figure 3-8 are used for model updating, i.e. $n_{\text{modes}} = 4$. As shown in Section 2.4, the Levenberg-Marquardt algorithm is always applicable to both modal property difference and modal dynamic residual formulations, i.e. Case 1(a), 2(a) and 3(a) in Table 2-1 are all applicable. On the other hand, the trust-region-reflective algorithm is applicable only if the length of residual vector is no less than the number of optimization variables, i.e. $m \geq n_{\mathbf{x}}$ in Eq. (2-33). For the MAC value formulation denoted as Case 1(b) in Table 2-1, Eq. (2-35) shows that the length of the residual vector, m equals $2 \cdot n_{\text{modes}} = 8$, while the number of optimization variables ($n_{\mathbf{x}}$) equals $n_{\boldsymbol{\alpha}} = 4$, and thus, $n_{\mathbf{x}} < m$. When using eigenvector difference formulation to update the FE model (Case 2(b) in Table 2-1), Eq. (2-36) presents that the length of the residual vector m equals $n_m \cdot n_{\text{modes}} = 16$. The number of optimization variable $n_{\mathbf{x}}$ still equals $n_{\boldsymbol{\alpha}} = 4$, making the optimization problem not underdetermined. For this structure, all the DOFs are instrumented with sensor, so number of unmeasured DOFs, $n_{\mathbf{u}}$, equals zero. Thus, the number of optimization variables ($n_{\mathbf{x}}$) for the modal dynamic residual formulation in Eq. (2-37) still equals $n_{\boldsymbol{\alpha}} = 4$. With the length of the residual vector, m,

equal to $N \cdot n_{\text{modes}} = 16$, m is larger than $n_{\mathbf{x}}$. Therefore, Case 1(b), 2(b) and 3(b) are all applicable for this shear-frame structure.

When using MATLAB `lsqnonlin` with the trust-region-reflective algorithm, the upper and lower bounds of $\boldsymbol{\alpha}$ are simply set to be 1 and -1, respectively. For each applicable case in Table 2-3, the optimization process is initiated from 100 random starting points of $\boldsymbol{\alpha}$, which are uniformly randomly generated between the upper and lower bounds. On the other hand, when using the Levenberg-Marquardt algorithm with MATLAB `lsqnonlin`, upper and lower bounds cannot be handled by the toolbox, as described in Section 2.4. Consequently, optimal result sets obtained from Levenberg-Marquardt algorithm that are out of the bounds are discarded and not included in the final result sets. Instead, the starting point is replaced with the next randomly generated point that can conclude the search within the desired bounds of optimization variables. As a result, for Case 1(a), 2(a) and 3(a) that use Levenberg-Marquardt algorithm, in order to obtain 100 sets of optimal results that are within the bounds, the Levenberg-Marquardt algorithm may have to solve the optimization problems from more than 100 starting points.

3.2.3.1 Updating Results of MAC Value Formulation

For this shear-frame structure, both Levenberg-Marquardt and trust-region-reflective algorithm can be applied on the MAC value formulation, i.e. Case 1(a) and 1(b) in Table 2-1. Analytical gradient (Eq. (2-14)) is used during the optimization process. Same as CHAPTER 2, among the 100 in-bound optimal result sets, the result set achieving the minimum objective function value is selected as the best solution. For the best solutions from two optimization algorithms, Figure 3-10 plots the optimal value of each updating

parameter (α_i^*). The figure shows that the optimal results from both algorithms are identical, with objective function value equal to 1.8480×10^{-5}. In addition, the figure also demonstrates that the inter-story stiffness value of lower stories is much lower than the top story, which is because of the P-Δ effect from the weight of aluminum plates. Table 3-4 summarizes the modal properties generated by the FE model built with α_i^* displayed in Figure 3-10. Since both Levenberg-Marquardt and trust-region-reflective algorithms provide a same set of optimal solution, only one updated FE model is included in Table 3-4. Compared to Table 3-2, the difference between the resonance frequencies and mode shapes becomes much smaller.

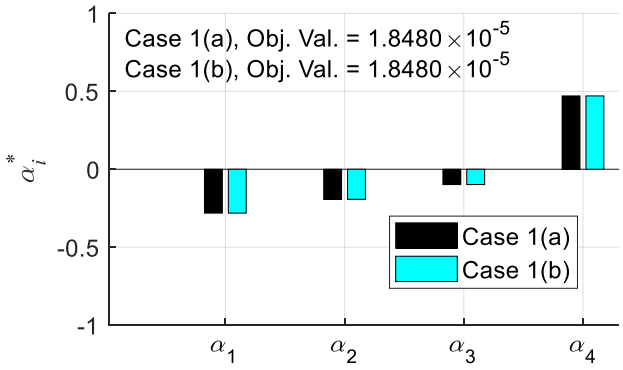


Figure 3-10. Shear frame structure: updating results of MAC value formulation using analytical gradient

Table 3-4 Comparison of experimental and simulated modal properties from MAC value formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.88	0.006	0.999
2nd mode	2.75	2.75	0.009	0.994
3rd mode	4.30	4.30	0.006	0.997
4th mode	5.53	5.53	0.010	0.995

3.2.3.2 Updating Results of Eigenvector Difference Formulation

Denoted as Case 2(a) and 2(b) in Table 2-1, both Levenberg-Marquardt and trust-region-reflective algorithms are applicable for the eigenvector difference formulation on this shear-frame structure. Similar to the MAC value formulation, analytical gradient shown in Eq. (2-25) is used during the optimization process. Among 100 in-bound optimal result sets, the best solution is selected based on the minimum objective function value. The optimal updating parameter values of two best solution sets are plotted in Figure 3-11. The figure demonstrates that Case 2(a) and 2(b) obtain the same optimal value with objective function value equal to 0.0390. Similar to the best result set of MAC value formulation, the inter-story stiffness value of the top story is the highest. Using the optimal value shown in Figure 3-11, an updated FE model can be built and used to generate simulated modal properties. Table 3-5 compares the simulated modal properties from the updated FE model with the experimental ones. Although the simulated modal properties get closer to the experimental ones, the difference is slightly larger than the optimal result set from the MAC value formulation shown in Table 3-4, especially for the difference in

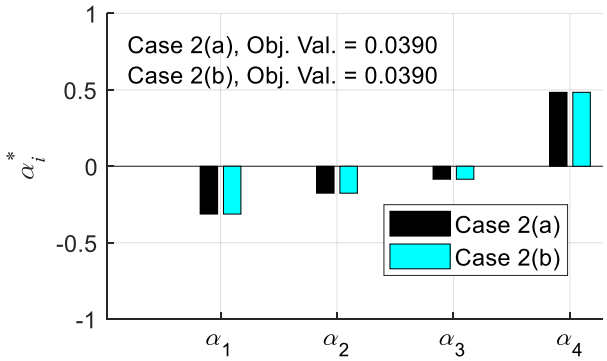


Figure 3-11. Shear frame structure: updating results of eigenvector difference formulation using analytical gradient

resonance frequencies. Nevertheless, such small differences are expected of the different formulations, and have little influence from the engineering point of view.

Table 3-5 Comparison of experimental and simulated modal properties from eigenvector difference formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.88	0.660	0.999
2nd mode	2.75	2.74	0.385	0.994
3rd mode	4.30	4.32	0.439	0.998
4th mode	5.53	5.56	0.578	0.994

3.2.3.3 Updating Results of Modal Dynamic Residual Formulation

As mentioned before, each story of this shear-frame structure is instrumented with sensor. Therefore, the optimization variables in the modal dynamic residual formulation in Eq. (2-10) contain the vector variable $\boldsymbol{\alpha}$ only. The modal dynamic residual formulation degenerates accordingly to a convex optimization problem. As a result, both Levenberg-Marquardt and trust-region-reflective algorithms are expected to converge at the same optimal solution point, regardless of the starting values of updating variables. Figure 3-12 plots the optimal values of updating variables, where the inter-story stiffness of the top story is again higher than other stories. As expected, the optimal result sets from the Levenberg-Marquardt and trust-region-reflective algorithms are the same, with the objective function value equal to 10.3809. An updated FE model is then constructed using the optimal updating variable values shown in Figure 3-12. The comparison between the experimental and simulated modal properties from the updated FE model is summarized

in Table 3-6, where obvious reduction of the difference in modal properties can be observed, i.e. lower Δf_i and higher MAC values compared to Table 3-2.

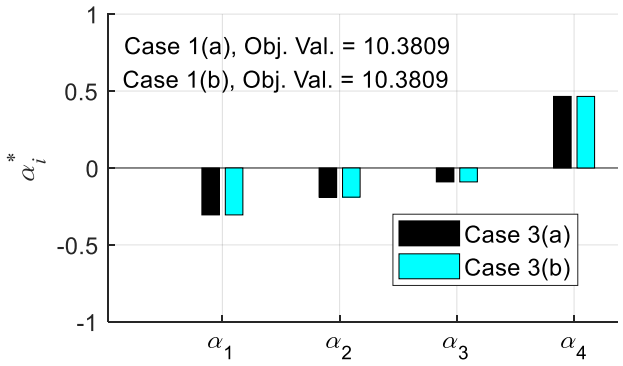


Figure 3-12. Shear frame structure: updating results of modal dynamic residual formulation using analytical gradient

Table 3-6 Comparison of experimental and simulated modal properties from modal dynamic residual formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.88	0.706	0.999
2nd mode	2.75	2.74	0.450	0.994
3rd mode	4.30	4.29	0.133	0.998
4th mode	5.53	5.53	0.037	0.994

3.2.3.4 Summary of FE Model Updating Results

Figure 3-10, Figure 3-11 and Figure 3-12 show that the optimal values of updating variables for three model updating formulations are different from each other. Columns in Table 3-7 summarize the nominal and optimal stiffness parameter values from each formulation, where Eq. (2-6), Eq.(2-8) and Eq. (2-10) represent the MAC value, eigenvector difference and modal dynamic residual formulation, respectively. Using the

nominal and three sets of optimal stiffness values shown in the table, four FE models are built, i.e. the initial model and three updated models. The objective function values of three model updating formulations are then evaluated using modal properties generated by each model and listed in Table 3-7. As intended, each model updating formulation minimizes its objective function, i.e. MAC value formulation minimizes Eq. (2-6); eigenvector difference formulation minimizes Eq.(2-8); modal dynamic residual formulation minimizes Eq. (2-10). Also, Table 3-7 shows that the achieved minimum values of the objective functions for all three formulations decrease significantly from the values of the nominal model.

Table 3-7 Comparison of optimal stiffness values among different model updating formulations

		Nominal	Eq. (2-6) (MAC value)	Eq.(2-8) (Eigvec. Diff.)	Eq. (2-10) (Modl. Dync. Res.)
Optimal spring stiffness values (lbf/in)	k_1	10	7.1868	6.8728	6.9494
	k_2	10	8.0725	8.2384	8.1029
	k_3	10	9.0180	9.1507	9.0942
	k_4	10	14.7048	14.8286	14.6502
Objective function value	Eq. (2-6) (MAC value)	0.1404	1.85×10^{-5}	4.62×10^{-4}	3.05×10^{-4}
	Eq.(2-8) (Eigvec. Diff.)	1.8674	4.13×10^{-2}	3.90×10^{-2}	3.95×10^{-2}
	Eq. (2-10) (Modl. Dync. Res.)	271.97	10.5482	10.8868	10.3809

Finally, Table 3-4, Table 3-5 and Table 3-6 demonstrate the updated model from the MAC value formulation generates resonance frequencies closest to the experimental

results. Such difference is expected of different model updating formulations. In terms of mode shapes, all three model updating formulations demonstrate similar performance. In the meantime, the difference among optimal stiffness values among columns shown in Table 3-7 is not significant from the engineering point of view. Last but not least, it is worth mentioning that when solving the optimization problems of three model updating formulations, the same setting is used in MATLAB `lsqnonlin` toolbox.

3.3 FE Model Updating with Field Experiment Data

On the four-story shear frame structure, *Martlet* wireless sensing node and the integrated accelerometer *wing* demonstrate comparable performance with respect to the high-precision cabled sensing system. In this subsection, the *Martlet* wireless sensing nodes were installed on a pedestrian bridge on Georgia Tech campus, as shown in Figure 3-13. The main frame of the pedestrian bridge is made of steel, and the bridge deck is made of concrete.

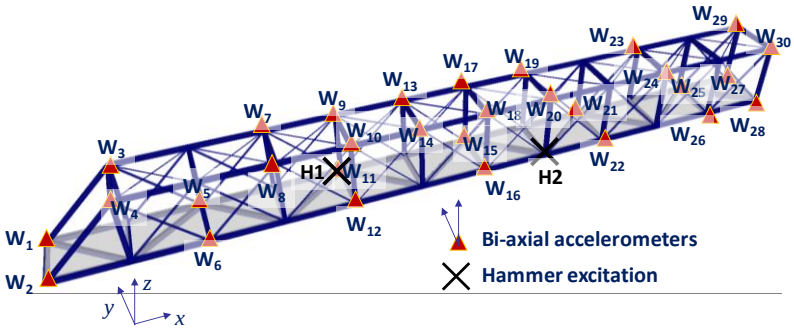
Figure 3-13. Pedestrain bridge on Georgia Tech campus

To collect structural vibration data, the integrated accelerometer *wing* is again connected with *Martlet* node. To obtain the frequency-domain modal properties, i.e.

resonance frequencies and mode shapes, of the pedestrian bridge, hammer impact is applied at designated locations on the bridge. Later, the extracted modal properties are used to update the FE model of the pedestrian bridge.

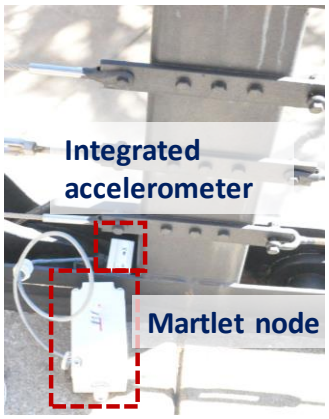
3.3.1 Test Configuration and Test Results

Figure 3-14(a) shows the instrumentation plan of the wireless sensors. In total 30 wireless sensors (W_1~ W_{30}) are installed on the top and bottom levels of the pedestrian bridge. At each location, one *Martlet* node with one integrated accelerometer *wing* are installed at the beam-column joint. Figure 3-14(b) shows a wireless sensor (W_{24}) installed



(a) Wireless sensor instrumentation plan

(b) Wireless sensor intalled on the top level of the bridge (W_{24})



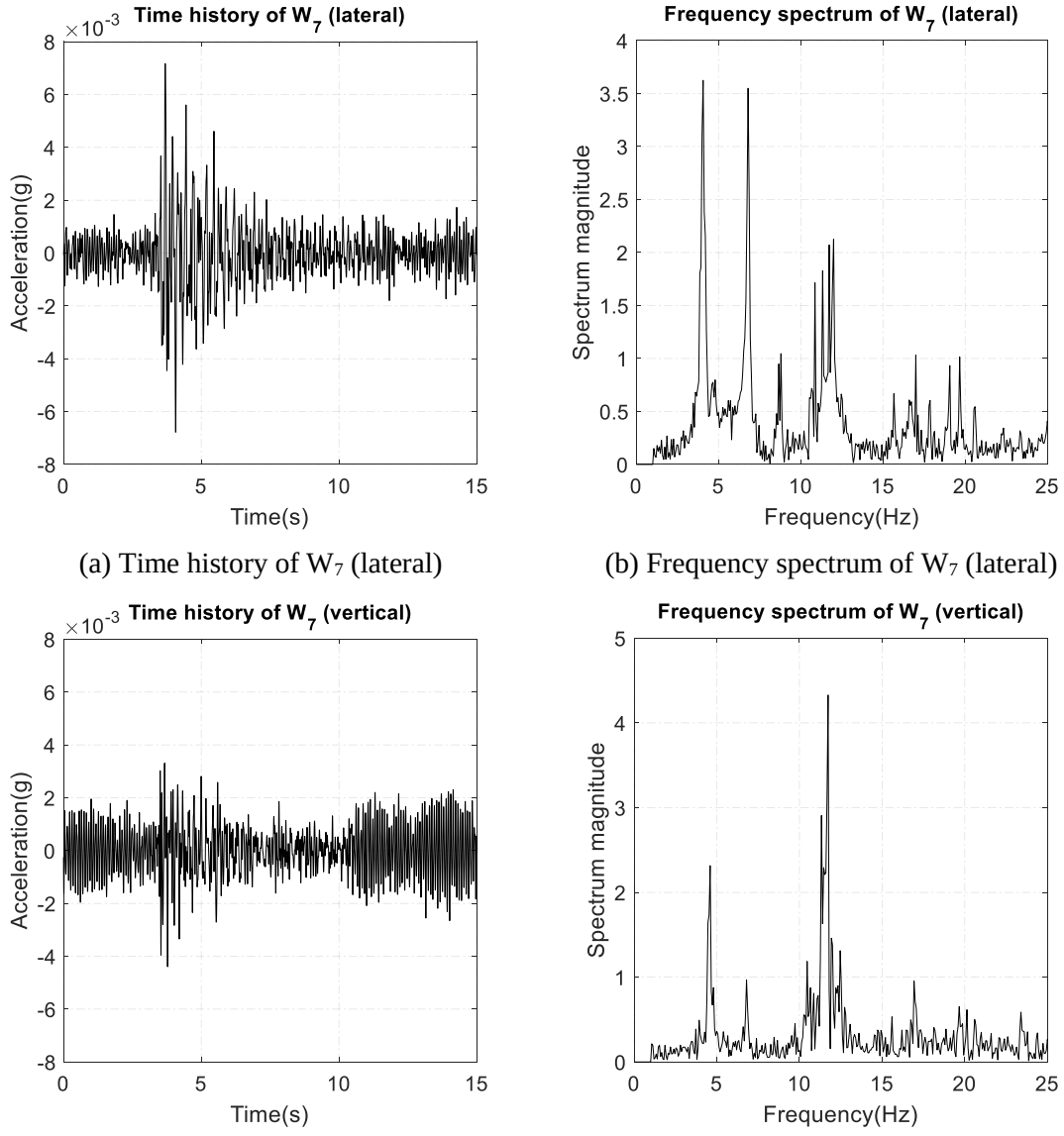
(c) Wireless sensor installed on the bottom level of the bridge (W_{22})

Figure 3-14. Experimental setup for the wirless sensor testing

at a joint on the top level. Figure 3-14(c) show a wireless sensor (W_{22}) installed at a joint on the bottom level.

To obtain the frequency-domain modal properties of the pedestrian bridge from acceleration data, a 12.1-lb hammer (PCB Piezotronics 086D50) is used to generate excitations on the bridge deck. The hammer impact is applied on two designated locations on the bridge deck, denoted as H_1 and H_2 in Figure 3-14(a). For this pedestrian bridge, we are interested in the vertical (along *z* axis) and lateral (along *y* axis) vibration modes. At each location, the hammer excitations are applied only vertically and laterally at the beam-column joint on the bottom level. Only the two axes of the integrated accelerometer *wing* are used to collect vibration data. To obtain better experimental modal properties for FE model updating, hammer impact experiments are repeated multiple times along both lateral and vertical direction at each location. In the following experiments, the amplification gain of the integrated accelerometer *wing*s is set to ×20, and the cutoff frequency is set as 25Hz. The sampling frequency of *Martlet* node is set as 1,000Hz.

Figure 3-15 plots two sets of time history data collected by *Martlet* node, as well as the corresponding frequency spectrum. The data is collected when hammer impact is applied laterally (along *y* direction) at H_1. Figure 3-15(a) and (c) show the acceleration data of lateral and vertical channel from sensor W_7. Since the hammer hits along lateral direction, the magnitude of the response signal from the lateral channel is larger than that from the vertical channel. In addition, the peaks in the frequency spectrum shown in Figure 3-15(b) and (d) may represent the resonance frequencies of this pedestrian bridge.



(a) Time history of W_7 (lateral)

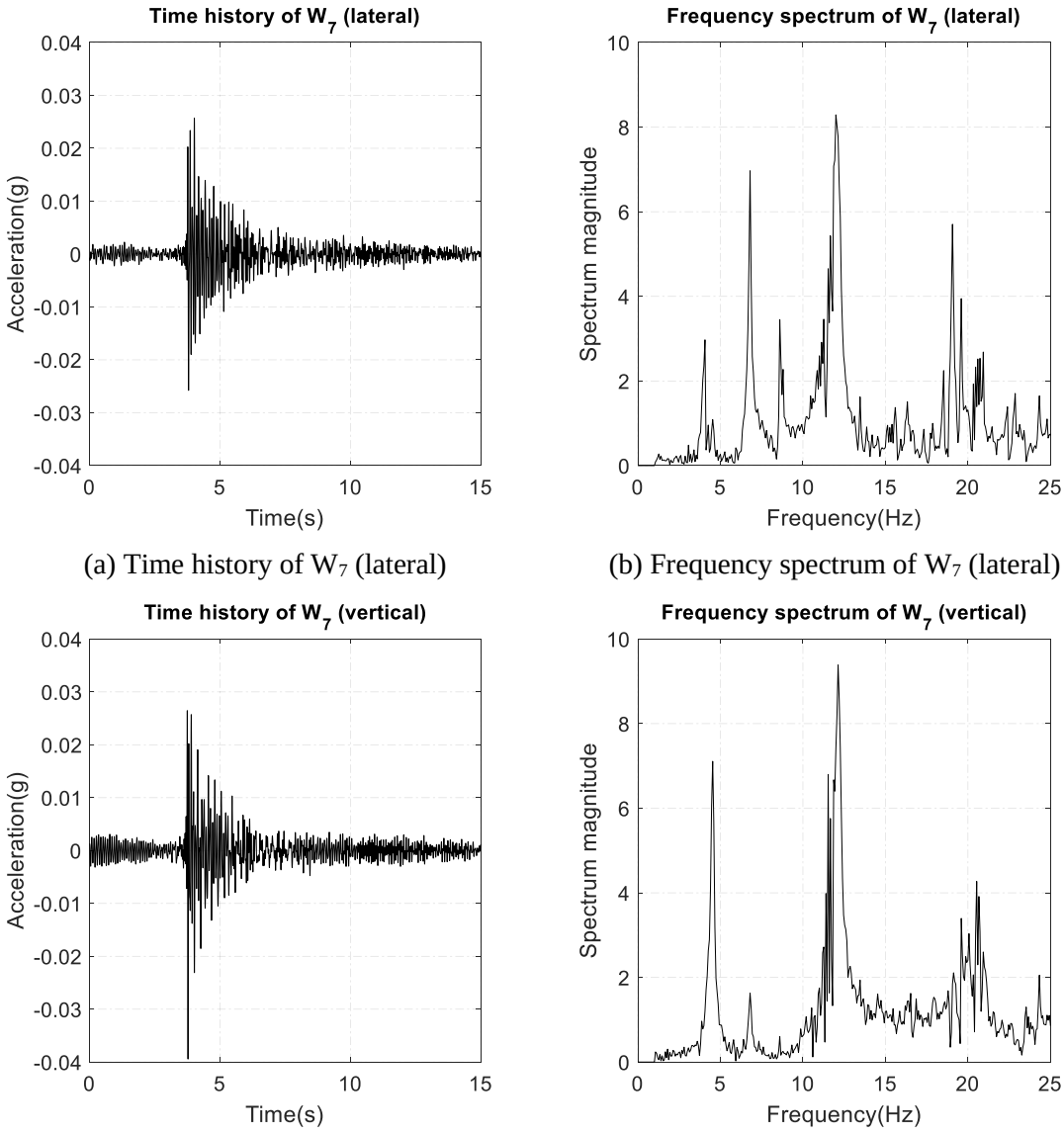
(b) Frequency spectrum of W_7 (lateral)

(c) Time history of W_7 (vertical)

(d) Frequency spectrum of W_7 (vertical)

Figure 3-15. Example vibration records and corresponding frequency spectra when hammer impact is applied laterally at H1

On the other hand, Figure 3-16 shows the time history data collected when hammer hits vertically (along *z* direction) at H_1. As expected, the magnitude of response data from the vertical channel, shown in Figure 3-16(c), is larger than that from the lateral channel, shown in Figure 3-16(a). In addition, the peaks in the frequency spectrum in Figure 3-16(b) and (d) agree with the peaks shown in Figure 3-15(b) and (d).



(a) Time history of W_7 (lateral) (b) Frequency spectrum of W_7 (lateral)

(c) Time history of W7 (vertical) (d) Frequency spectrum of W7 (vertical)

Figure 3-16. Example vibration records and corresponding frequency spectra when hammer impact is applied vertically at H1

3.3.2 Modal Analysis Results

To use the model updating formulation shown in CHAPTER 2 to update the FE model of the pedestrian bridge, the modal properties of the bridge are extracted from the collected acceleration data. For this pedestrian bridge, Numerical Algorithms for Subspace State Space System Identification (N4SID) is adopted to extract the modal properties [64].

The N4SID algorithm first identifies the state space model of the pedestrian bridge with the collected acceleration data. Then, by solving the eigenvalue problem of the identified state space model, the modal properties of the pedestrian bridge, i.e. experimental eigenvalues and eigenvectors, can be extracted. However, the eigenvectors obtained from the N4SID algorithm is in complex domain. Meanwhile, the model updating formulations presented in CHAPTER 2 only work with real-valued eigenvectors. Therefore, this subsection will first propose a method to generate real-valued vectors that approximate the complex-valued eigenvectors.

Furthermore, as stated in Section 3.3.1, hammer impact experiments are repeated multiple times along both lateral and vertical directions at each location. As a result, multiple sets of experimental modal properties can be extracted from the hammer impact experiments, while the model updating formulations presented in CHAPTER 2 only requires one set of experimental modal properties. Thus, this subsection will also present a method to generate a resultant set of modal properties from multiple modal analysis results.

3.3.2.1 Complex-Valued Eigenvector Approximation

Recall that n_{m} represents the number of measured DOFs, and the *i*-th complex-valued eigenvector, $\boldsymbol{\psi}_i^{\mathrm{cplx}} \in \mathbb{C}^{n_{\mathrm{m}}}$, obtained from the N4SID algorithm can be defined as follows.

$$\boldsymbol{\psi}_i^{\mathrm{cplx}} = \boldsymbol{\psi}_i^{\mathrm{real}} + j \cdot \boldsymbol{\psi}_i^{\mathrm{imag}}, i = 1 \dots n_{\mathrm{modes}} \tag{3-1}$$

where $\boldsymbol{\psi}_i^{\text{real}}, \boldsymbol{\psi}_i^{\text{imag}} \in \mathbb{R}^{n_\text{m}}$ represent the real part and imaginary part of $\boldsymbol{\psi}_i^{\text{cplx}}$, respectively; j stands for the imaginary unit; n_{modes} is the number of the identified modes from the N4SID algorithm. At the first step, the complex eigenvector is normalized so that the same entry (q_i-th entry) in the real and imaginary part equals 1, i.e. $\psi_{q_i,i}^{\text{real}} = \psi_{q_i,i}^{\text{imag}} = 1$. As stated in [74], for lightly damped structure, the collinearity between the real and imaginary part of the complex-valued eigenvector is supposed to be high. By assuming the pedestrian bridge is a lightly damped structure, after normalization, entries in $\boldsymbol{\psi}_i^{\text{cplx}}$ would lie close to the 45° line in the complex domain. Therefore, if an entry deviates away from the 45° line, we consider the accuracy of the entry to be low. These entries in $\boldsymbol{\psi}_i^{\text{cplx}}$ with low collinearity between the real and imaginary parts will be removed from $\boldsymbol{\psi}_i^{\text{cplx}}$. The coefficient of determination [75] between real and imaginary parts of $\boldsymbol{\psi}_i^{\text{cplx}}$, $\text{R}^2 \in \mathbb{R}$, with respect to the 45° line can be defined as follows:

$$\text{R}^2 = 1 - \frac{\sum_{k=1}^{n_\text{m}} \left(\psi_{k,i}^{\text{real}} - \psi_{k,i}^{\text{imag}}\right)^2}{\sum_{k=1}^{n_\text{m}} \left(\psi_{k,i}^{\text{imag}} - \bar{\psi}_i^{\text{imag}}\right)^2} \tag{3-2}$$

where $\bar{\psi}_i^{\text{imag}}$ represents the mean value of $\boldsymbol{\psi}_i^{\text{imag}}$. When R^2 value is close 1, $\boldsymbol{\psi}_i^{\text{real}}$ and $\boldsymbol{\psi}_i^{\text{imag}}$ are highly collinear. In the proposed method, if R^2 value is smaller than the prescribed threshold value, the point in $\boldsymbol{\psi}_i^{\text{cplx}}$ that deviates most from the 45° line will be first removed from $\boldsymbol{\psi}_i^{\text{cplx}}$. The points in $\boldsymbol{\psi}_i^{\text{cplx}}$ keep being removed until the R^2 value of the remaining points is higher than the threshold value. Finally, points on the 45° line are used to approximate the remaining points in $\boldsymbol{\psi}_i^{\text{cplx}}$. Using a low-dimensional vector in

complex domain, Figure 3-17 illustrates the process of approximating a complex-valued vector, $\boldsymbol{\psi}_i^{\text{cplx}} \in \mathbb{C}^5$ with the points, $\boldsymbol{\psi}_i^{\text{fit}} + j \cdot \boldsymbol{\psi}_i^{\text{fit}} \in \mathbb{C}^5$ on the 45° line. The asteroid and the circle points in the figure represent the points of the original vector ($\boldsymbol{\psi}_i^{\text{cplx}}$) and the approximation points ($\boldsymbol{\psi}_i^{\text{fit}}$), respectively. It can be seen from the objective is to project the points of the original vector onto the 45° line.

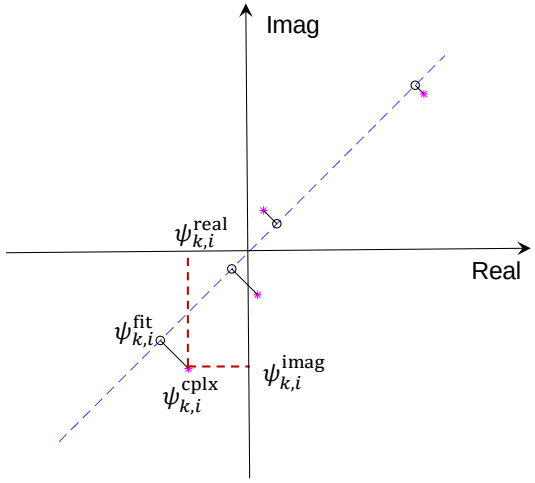


Figure 3-17. Illustration of approximating a complex-valued vector with a real-valued vector

In order to find those approximation points, a series of optimization problems are to be solved.

$$\underset{\psi_{k,i}^{\text{fit}}}{\text{minimize}} \quad \left(\psi_{k,i}^{\text{real}} - \psi_{k,i}^{\text{fit}}\right)^2 + \left(\psi_{k,i}^{\text{imag}} - \psi_{k,i}^{\text{fit}}\right)^2 \tag{3-3}$$

where $\psi_{k,i}^{\text{fit}} \in \mathbb{R}$ is the only optimization variable of the *k*-th optimization problem. Each optimization problem represented as Eq. (3-3) is convex with respect to $\psi_{k,i}^{\text{fit}}$, whose

solution can be easily calculated. Eq. (3-3) aims to find a point on the 45° line, i.e. $\left(\psi_{k,i}^{\text{fit}}, \psi_{k,i}^{\text{fit}}\right)$, with the shortest distance to the original point, i.e. $\left(\psi_{k,i}^{\text{real}}, \psi_{k,i}^{\text{imag}}\right)$.

After solving Eq. (3-3) for all the remaining points in $\boldsymbol{\psi}_i^{\text{cplx}}$, a real-valued vector ($\boldsymbol{\psi}_i^{\text{fit}}$) can be obtained to approximate the complex-valued eigenvector. Finally, it is worth noting that with $\psi_{q_i,i}^{\text{real}} = \psi_{q_i,i}^{\text{imag}} = 1$, the q_i-th entry in $\boldsymbol{\psi}_i^{\text{fit}}$ still equals 1, i.e. $\psi_{q_i,i}^{\text{fit}} = 1$.

3.3.2.2 Resultant Modal Analysis Results

So far, we successfully retrieve multiple sets of experimental eigenvalues and real-valued eigenvector from the collected acceleration measurements on the pedestrian bridge. In this subsection, the objective is to generate a resultant set of modal properties from multiple modal analysis results. The calculation of resultant eigenvalue is relatively easy. Assuming that the number of *i*-th eigenvalues identified from hammer impact experiments is n_i, the resultant *i*-th eigenvalue, $\bar{\lambda}_i^{\text{EXP}} \in \mathbb{R}$, can be expressed as below:

$$\bar{\lambda}_i^{\text{EXP}} = \frac{\sum_{l=1}^{n_i} \lambda_i^l}{n_i}, i = 1 \dots n_{\text{modes}} \tag{3-4}$$

where $\lambda_i^l \in \mathbb{R}$, represents the i-th eigenvalue from the l-th modal analysis result. Eq. (3-4) shows that the resultant i-th eigenvalue, $\bar{\lambda}_i^{\text{EXP}}$, is simply the mean value of all the identified eigenvalues of the i-th mode from the hammer impact experiments.

In the next step, a resultant set of eigenvectors will be calculated. Recall that when approximating the complex-valued eigenvector, the entries in $\boldsymbol{\psi}_i^{\text{cplx}}$ with low collinearity between the real and imaginary part are removed. Thus, different entries in the i-th

experimental eigenvector have different number of analysis results from hammer experiments. Here we use $n_{k,i}$ to denote the number of analysis results for the k-th entry in the i-th eigenvector. It is worth noting that $0 \leq n_{k,i} \leq n_i$. The k-th entry in the i-th resultant eigenvector, $\bar{\psi}_{k,i}^{\mathrm{EXP,m}} \in \mathbb{R}$, is defined as the weighted mean value of all the identified k-th entry in the i-th real-valued eigenvector:

$$\bar{\psi}_{k,i}^{\mathrm{EXP,m}} = \frac{\sum_{l=1}^{n_{k,i}} \psi_{k,i}^{l,\mathrm{fit}} \cdot w_{k,i}^{l}}{\sum_{l=1}^{n_{k,i}} w_{k,i}^{l}}, i = 1 \dots n_{\mathrm{modes}};\ k = 1 \dots n_{\mathrm{m}} \tag{3-5}$$

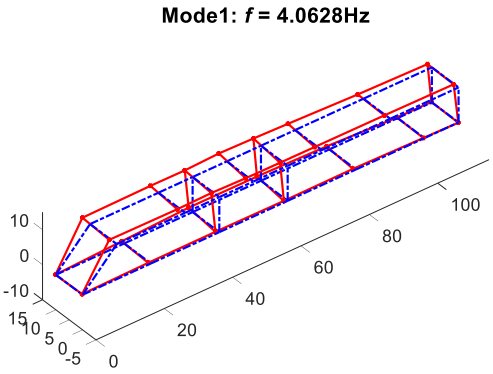
where $\psi_{k,i}^{l,\mathrm{fit}}$ denotes the k-th entry in the i-th real-valued eigenvector from the l-th identification result; $w_{k,i}^{l}$ is the weighting factor of $\psi_{k,i}^{l,\mathrm{fit}}$. The definition of $w_{k,i}^{l}$ is shown below:

$$w_{k,i}^{l} = 1 - \frac{\left(\psi_{k,i}^{l,\mathrm{fit}} - \psi_{k,i}^{l,\mathrm{real}}\right)^2 + \left(\psi_{k,i}^{l,\mathrm{fit}} - \psi_{k,i}^{l,\mathrm{imag}}\right)^2}{\left(\psi_{k,i}^{l,\mathrm{real}}\right)^2 + \left(\psi_{k,i}^{l,\mathrm{imag}}\right)^2} \tag{3-6}$$

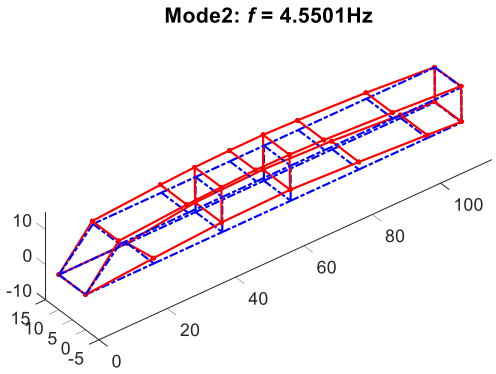
where $\psi_{k,i}^{l,\mathrm{real}}$ and $\psi_{k,i}^{l,\mathrm{imag}} \in \mathbb{R}$ represent the k-th entry in $\boldsymbol{\psi}_i^{\mathrm{real}}$ and $\boldsymbol{\psi}_i^{\mathrm{imag}}$ from the l-th analysis result. Eq. (3-6) implies that the larger the error of the approximated real-valued eigenvector, the smaller the weighting factor is. It is also worth noting that with the q_i-th entry in each $\boldsymbol{\psi}_i^{l,\mathrm{fit}}$ always equal to 1, the q_i-th entry in the resultant eigenvector equals 1 as well, i.e. $\bar{\psi}_{q_i,i}^{\mathrm{EXP}} = 1$.

Using the method introduced above, a set of resultant modal properties of the pedestrian bridge are extracted from the collected structural response to hammer impacts.

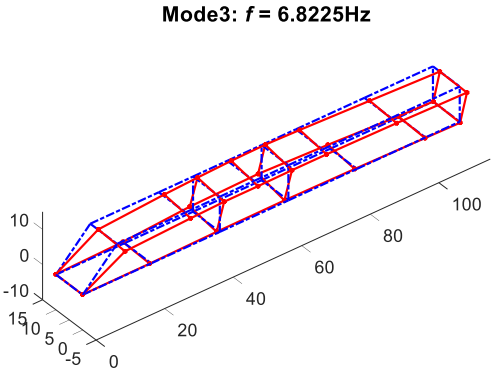
During the modal analysis, two measurement channels are found to be malfunctioning, i.e. vertical channel of W_5 and W_{29}. Thus, in total 58 DOFs of the steel pedestrian bridge was successfully measured during the hammer impact experiments. Furthermore, when removing unreliable channels from the complex eigenvector, the threshold value of coefficient of determination is set as 0.90. Figure 3-18 plots the first five resultant modal properties of the pedestrian bridge. The figure shows that the 1st and 3rd mode are vibrating laterally (along *y* direction), while the 2nd and 5th modes are vibrating vertically (along *z* direction). The 4th mode is a torsional mode. Finally, the resonance frequencies of the resultant modal properties are close to the peaks shown in Figure 3-15 and Figure 3-16.



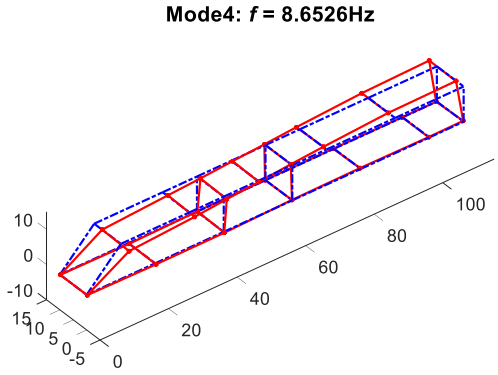
(a) Lateral I mode



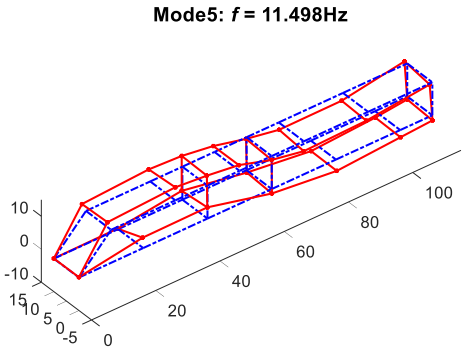
(b) Vertical I mode



(c) Lateral II mode



(d) Torsinal I mode



(e) Vertical II mode

Figure 3-18. First four resultant modal proeprties of the pedestrain bridge

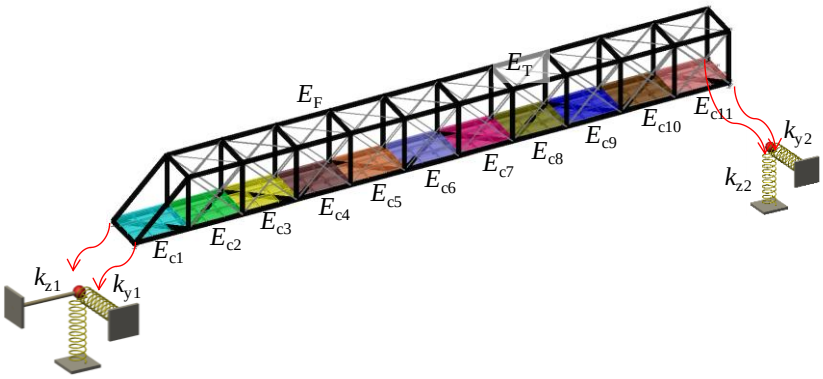
3.3.3 FE Model Updating of the Pedestrian Bridge

This subsection aims to use the experimental modal properties obtained from the previous subsection to update the FE model of the pedestrian bridge. First, a preliminary FE model of the bridge is constructed based on the design drawings. Second, the MAC value and eigenvector difference formulations are adopted to update the selected stiffness parameters of the pedestrian bridge model.

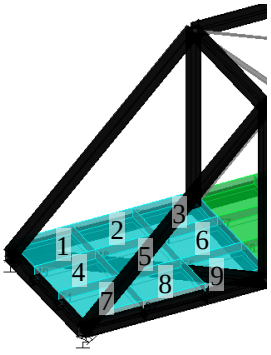
3.3.3.1 Preliminary FE Model of the Pedestrian Bridge

A preliminary FE model of the pedestrian bridge is constructed in SAP2000, as shown in Figure 3-19(a). The main frame of the bridge is modelled as frame elements with steel material. The segmental diagonal bracings in top plane and two side planes in the FE model are constructed with truss elements. The deck of the bridge is built with concrete material and modelled as shell-thin element. As shown in the design drawing, the concrete deck is connected with the bottom frame members with shear studs. Therefore, the shell elements and frame elements on the bottom plane are sharing nodes in the FE model. Figure

3-19(b) shows a close-up view of the 1st concrete segment of the bridge model. The concrete slab in the segment is meshed into nine elements in a 3×3 formation. Transverse and vertical springs (k_y and k_z) are allocated at both ends of the bridge to simulate non-ideal boundary conditions. The longitudinal (x-direction) DOF of two left end nodes are constrained. In total, the FE model has 946 DOFs. Similar to the numerical examples shown in CHAPTER 2, the mass matrix (**M**) is a diagonal matrix whose diagonal entries associated with rotational DOFs equal zero.



(a) Overview of the FE model



(b) Close-up view of the concrete elements in the first segment

Figure 3-19. FE model of the pedestrian bridge

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Table 3-8 lists the stiffness parameters to be updated. The Young's moduli of all the frame elements are grouped as one parameter (E_{F}). The Young's moduli of all the truss members are also grouped as one parameter (E_{T}). As shown in Figure 3-19(a), the Young's modulus of concrete elements in each segment is updated separately ($E_{\mathrm{c1}} \sim E_{\mathrm{c11}}$). The updated parameters also include the stiffness values of support springs (k_{y1}, k_{z1}, k_{y2} and k_{z2}). Table 3-8 lists the nominal values of the stiffness parameters. Similar to CHAPTER 2, $\boldsymbol{\alpha}$ is scaled to represent the relative change percentage from nominal value of each stiffness parameter. In total, this model updating problem has 17 updating variables, i.e. $n_{\boldsymbol{\alpha}} = 17$. The last column in Table 3-8 lists the lower and upper bounds for each of $\boldsymbol{\alpha}$. The variation

Table 3-8 Structural properties of the pedestrian bridge

Stiffness parameters		Nominal value	Updating variables	$[L_\alpha, U_\alpha]$
Elastic moduli of frame elements (kips/in^2)	E_{F}	29,000	α_1	[-0.1, 0.1]
Elastic moduli of truss elements (kips/in^2)	E_{T}	29,000	α_2	[-0.1, 0.1]
Elastic moduli of concrete elements (kips/in^2)	$E_{\mathrm{c1}} \sim E_{\mathrm{c11}}$	3122.02	$\alpha_3 \sim \alpha_{13}$	[-0.2, 0.4]
Support spring stiffness (kips/in)	k_{y1}	400	α_{14}	[-1, 10]
	k_{z1}	500	α_{15}	
	k_{y2}	400	α_{16}	
	k_{z2}	500	α_{17}	

of the steel material is expected to be smaller than that of the concrete material. The bounds for the updating variables of steel members (α_1 and α_2) is therefore tighter than the concrete members (α_3~α_{13}). Among all the stiffness parameters, the stiffness values of support springs are most difficult to estimate. Therefore, the bounds for $\alpha_{14} \sim \alpha_{17}$ are the loosest.

Table 3-9 compares the experimental modal properties shown in Figure 3-18 and simulated modal properties generated from the nominal FE model. The MAC value is as defined in Eq. (2-7). Except for the 5th mode, obvious difference can be observed in the resonance frequencies. The MAC values for the 1st and 2nd modes are very low, which implies the low collinearity between the experimental and simulated mode shapes. Upon inspecting the mode shapes, it is found that the mode sequence of the 1st and 2nd simulated mode shapes is different from the experimental ones. In other words, Figure 3-18(a) shows that the 1st experimental mode shape vibrates laterally, while the 1st simulated mode shape vibrates vertically. Similarly, the 2nd experimental mode shape vibrates vertically, and the 2nd simulated mode shape vibrates laterally.

Table 3-9 Comparison of experimental and simulated modal properties from the nominal model

Modes	Experimental results	Nominal model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	4.06	4.66	14.691	0.002
2nd mode	4.55	4.88	7.141	0.010
3rd mode	6.82	6.65	2.470	0.955
4th mode	8.65	9.90	14.427	0.966
5th mode	11.50	11.38	1.018	0.903

3.3.3.2 Updating Results of MAC Value Formulation

The MAC value formulation is first adopted to update the FE model of the pedestrian bridge. For real application, the MAC value formulation introduced in Section 2.2.1 is rewritten as follows:

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad \sum_{i=1}^{n_{\text{modes}}} \left\{ \left(\left(\bar{\lambda}_i^{\text{EXP}} - \lambda_i(\boldsymbol{\alpha}) \right) \cdot w_{\lambda_i} \right)^2 + \left(\frac{1 - \sqrt{\overline{\text{MAC}}_i(\boldsymbol{\alpha})}}{\sqrt{\overline{\text{MAC}}_i(\boldsymbol{\alpha})}} \cdot w_{\boldsymbol{\psi}_i} \right)^2 \right\} \tag{3-7a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{3-7b}$$

where $\bar{\lambda}_i^{\text{EXP}}$ is the i-th resultant experimental eigenvalue in Eq. (3-4); $\overline{\text{MAC}}_i(\boldsymbol{\alpha})$ is the MAC value between $\bar{\boldsymbol{\psi}}_i^{\text{EXP,m}}$in Eq. (3-5) and simulated eigenvector, $\boldsymbol{\psi}_i^{\text{m}}(\boldsymbol{\alpha})$; w_{λ_i} represent the weighting factor for the eigenvalue difference for the i-th mode; $w_{\boldsymbol{\psi}_i}$ is the weighting factor for the eigenvector difference for the i-th mode. For this pedestrian bridge model, the weighting factor for the i-th eigenvalue difference is calculated as follows:

$$w_{\lambda_i} = \frac{1}{\sigma_{\lambda_i}} \tag{3-8}$$

where σ_{λ_i} is the standard deviation of the identified i-th experimental eigenvalue from hammer impact experiments. Similarly, the weighting factor for the i-th eigenvector difference is formulated as below:

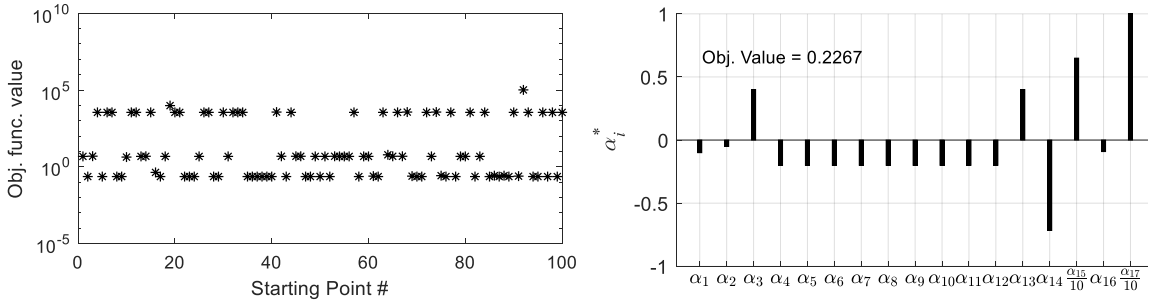
$$w_{\boldsymbol{\psi}_i} = \frac{1}{\left\| \boldsymbol{\sigma}_{\boldsymbol{\psi}_i} \right\|_2} \tag{3-9}$$

where $\|\cdot\|_2$ stands for a $\mathcal{L}_2$-norm of a vector; $\boldsymbol{\sigma}_{\boldsymbol{\Psi}_i} \in \mathbb{R}^{n_\mathrm{m}-1}$ contains the weighted standard deviation of $\bar{\boldsymbol{\Psi}}_i^{\mathrm{EXP}}$ in Eq. (3-5), except for the q_i-th entry. Since the q_i-th of all $\boldsymbol{\Psi}_i^{l,\mathrm{fit}}$ in Eq. (3-5) is equal to 1, the weighted standard deviation of the q_i-th entry in $\bar{\boldsymbol{\Psi}}_i^{\mathrm{EXP}}$ simply equals 0. Each entry in $\boldsymbol{\sigma}_{\boldsymbol{\Psi}_i}$ can be shown as below:

$$\sigma_{\boldsymbol{\Psi}_i}^k = \sqrt{\frac{n_{k,i}}{n_{k,i}-1} \cdot \frac{\sum_{l=1}^{n_{k,i}} w_{k,i}^l \cdot \left(\psi_{k,i}^{l,\mathrm{fit}} - \bar{\psi}_{k,i}^{\mathrm{EXP,m}}\right)^2}{\sum_{l=1}^{n_{k,i}} w_{k,i}^l}}, k = 1 \dots n_m, k \neq q_i \tag{3-10}$$

where $w_{k,i}^l$ is as defined in Eq. (3-6). In Section 3.3.2.2, five modes are obtained from the measured acceleration data, i.e. $n_{\mathrm{modes}} = 5$. As shown in Section 2.4, the trust-region-reflective algorithm is applicable only if the length of residual vector is no less than the number of optimization variables, i.e. $m \geq n_{\mathbf{x}}$ in Eq. (2-33). For the MAC value formulation shown in Eq. (3-7), the length of the residual vector, m equals $2 \cdot n_{\mathrm{modes}} = 10$, while the number of optimization variables ($n_{\mathbf{x}}$) equals $n_{\boldsymbol{\alpha}} = 17$, and thus, $n_{\mathbf{x}} < m$. The MAC value formulation cannot be solved by the trust-region-reflective algorithm for this pedestrian bridge model. On the other hand, although the Levenberg-Marquardt algorithm is applicable to the MAC value formulation, it does not allow to set the lower and upper bounds of the optimization variables. Therefore, in order to guarantee the optimal results are within the assigned bounds, the interior-point optimization algorithm implemented in the MATLAB `fmincon` solver is adopted [76]. Similar to the previous examples, the optimization process is initiated from 100 random starting points of $\boldsymbol{\alpha}$, which are uniformly randomly generated between the upper and lower bounds, i.e. the last column in Table 3-8.

Figure 3-20(a) plots the optimal objective function values of 100 random starting points when analytical gradient is used during the optimization process. The figure demonstrates that 35 optimization processes converge at the same optimal point. For the result set with the minimum objective function value among 100 result sets, Figure 3-20(b) plots the optimal value of each updating parameter (α_i^*). For display purpose, the optimal values of α_{15} and α_{17} shown in the figure are divided by 10. The figure shows that among all the updating variables, the updating variables related to vertical support spring stiffness, i.e. α_{15} for k_{z1} and α_{17} for k_{z2} increase mostly from the nominal value. With α_i^* displayed in Figure 3-20(b), an updated FE model can be built. Table 3-10 compares the modal properties generated by the updated FE model with the experimental modal properties. The resonance frequencies of the updated model is almost identical to the experimental resonance frequencies. Furthermore, the MAC values of the eigenvectors for the 1st and 2nd mode increase significantly, which implies that the mode sequence of the updated model is same as the experimental one. Although the MAC value for the 3rd, 4th and 5th mode decreases, the collinearity between the experimental and simulated eigenvectors is still



(a) Objective function values of 100 optimal result sets

(b) Updating parameter values from best result set

Figure 3-20. Pedestrian bridge model: updating results of MAC value formulation using analytical gradient

acceptable. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 5 hours and 24 minutes to obtain the 100 optimal result sets.

Table 3-10 Comparison of experimental and simulated modal properties from MAC value formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	4.06	4.06	0.009	0.934
2nd mode	4.55	4.56	0.150	0.980
3rd mode	6.82	6.82	0.054	0.940
4th mode	8.65	8.65	0.002	0.917
5th mode	11.50	11.43	0.585	0.884

3.3.3.3 Updating Results of Eigenvector Difference Formulation

Using the statistic weightings, the eigenvector difference formulation in Section 2.2.2 is reformulated as follows:

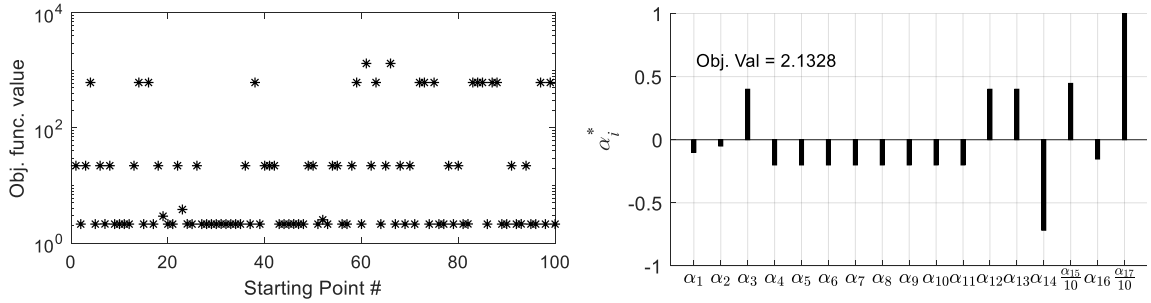
$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \quad \sum_{i=1}^{n_{\text{modes}}} \left\{ \left(\left(\bar{\lambda}_i^{\text{EXP}} - \lambda_i(\boldsymbol{\alpha}) \right) \cdot w_{\lambda_i} \right)^2 + \sum_{k=1, k\neq q_i}^{n_m} \left(\frac{\bar{\psi}_{k,i}^{\text{EXP}} - \psi_{k,i}^{m}(\boldsymbol{\alpha})}{\sigma_{\boldsymbol{\psi}_i}^{k} \cdot n_{\text{m}}} \right)^2 \right\} \tag{3-11a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{3-11b}$$

where the definition of $w_{\lambda_i} \in \mathbb{R}$ and $\sigma_{\boldsymbol{\psi}_i}^{k} \in \mathbb{R}$ is same as Eq. (3-8) and Eq. (3-9). To be consistent with the MAC value formulation, the interior-point optimization algorithm implemented in the `fmincon` solver is adopted. Again, in order to increase the chance of

finding the global minimum, 100 random starting points are generated to initiate the optimization process.

Still using the analytical gradient during the optimization process, Figure 3-21(a) plots the optimal objective function values of 100 result sets. The figure demonstrates that 48 optimization processes end the search at the same optimal point. Figure 3-21(b) plots the optimal value of each updating variables (α_i^*) for the best solution set. For display purpose, the optimal values of α_{15} and α_{17} shown in the figure are divided by 10. Similar to the results from MAC value formulation, α_{15} and α_{17} have the greatest increase. Using the optimal values shown in Figure 3-21(b), an updated FE model can be built. The comparison of the experimental modal properties and simulated modal properties from the updated model is summarized in Table 3-11. Compared with Table 3-9, the difference of resonance frequencies for all five modes decrease. As for the eigenvector comparison, similar to the result from the MAC value formulation, the increase of MAC value for the 1st and 2nd modes is significant, which implies the mode sequence of the simulated model is corrected during the model updating process. The MAC value of the rest of three modes decreases slightly from the nominal model, but the simulated eigenvectors from the



(a) Objective function values of 100 optimal result sets

(b) Updating parameter values from best solution set

Figure 3-21. Pedestrian bridge model: updating results of eigenvector difference formulation using analytical gradient

updated model is still close enough to the experimental eigenvectors. On the same PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 5 hours and 22 minutes to obtain the 100 optimal result.

Table 3-11 Comparison of experimental and simulated modal properties from eigenvector difference value formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	4.06	4.06	0.024	0.934
2nd mode	4.55	4.56	0.159	0.980
3rd mode	6.82	6.82	0.072	0.940
4th mode	8.65	8.65	0.002	0.918
5th mode	11.50	11.43	0.567	0.883

3.3.3.4 Summary of FE Model Updating Results

Figure 3-20(b) and Figure 3-21(b) demonstrate that the optimal values of the updating variables from the MAC value and eigenvector difference formulation are not the same. Table 3-12 summarizes the modal properties obtained from the experimental data, nominal model and two updated models, i.e. one from MAC value formulation and the other one from eigenvector difference formulation. The table shows that the modal properties of two updated models are much closer to the experimental modal properties than the nominal model. When comparing between two updated models, both simulated resonance frequencies and eigenvectors of the two updated models are close to each other. Figure 3-20(b) and Figure 3-21(b) reveal that the optimal values of α_{12} and α_{15} from two model updating formulations have large difference, which is expected of different model

updating formulations. For this pedestrian bridge model, since most of the stiffness is contributed from steel members, the variation of α_{12} (corresponding to concrete elastic modulus of E_{c10}) does not greatly affect the overall stiffness of the FE model. As for α_{15}, optimal values shown in two figures demonstrate that the vertical support spring (k_{z1}) is close to an ideal boundary condition. Therefore, the difference in optimal value of α_{15} does not significantly change the performance of the FE model. From the engineering point of view, the optimal FE models from two model updating formulations are close enough. Similarly, the modal properties shown in Table 3-12 can lead to the same conclusion.

Table 3-12 Comparison of updated model from different model updating formulations

Modes	Experimental results	Nominal model			Eq. (3-7) (MAC value)			Eq. (3-11) (Eigvec. Diff.)		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC	f_i (Hz)	Δf_i (%)	MAC	f_i (Hz)	Δf_i (%)	MAC
1st mode	4.06	4.66	14.691	0.003	4.06	0.009	0.934	4.06	0.023	0.934
2nd mode	4.55	4.88	7.141	0.010	4.55	0.150	0.980	4.55	0.159	0.980
3rd mode	6.82	6.65	2.470	0.955	6.82	0.054	0.939	6.82	0.072	0.940
4th mode	8.65	9.90	14.427	0.966	8.65	0.002	0.918	8.65	0.002	0.918
5th mode	11.50	11.38	1.018	0.903	11.43	0.585	0.889	11.43	0.567	0.883
Objective function value	Eq. (3-7) (MAC value)	5.57×10^3			0.2267			0.2436		
	Eq. (3-11) (Eigvec. Diff.)	2.02×10^4			2.1470			2.1328		

Last but not least, Table 3-12 also lists the objective function values of two model updating formulations evaluated with initial and two best solution sets. As intended, each model updating formulation minimizes its objective function, i.e. MAC value formulation

minimizes the objective function in Eq. (3-7) as 0.2267; eigenvector difference formulation minimizes the objective function in Eq. (3-11) as 2.1328. Also, Table 3-12 shows that the achieved minimum values of the objective functions for both formulations decrease significantly from these of the nominal model.

CHAPTER 4. FE MODEL UPDATING WITH GLOBAL OPTIMIZATION ALGORITHMS

As mentioned in CHAPTER 2, the optimization problems of MAC value, eigenvector difference and modal dynamic residual formulations, i.e. Eqs. (2-6), (2-8), (2-10) introduced in CHAPTER 2, generally are nonconvex with unknown number of local minima. Therefore, the local search optimization algorithms introduced in CHAPTER 2, i.e. Levenberg-Marquardt and trust-region-reflective algorithms cannot guarantee the global optimality of the optimal solutions. The numerical simulations presented in CHAPTER 2 also demonstrate the nonconvexity of all three model updating formulations.

To overcome the limitation of local search algorithms, this chapter will investigate two global optimization algorithms for solving the model updating problems. These are the branch-and-bound (B&B) algorithm [43] , and the primal-relaxed dual (P-RD) algorithm [45]. The MAC value formulation (Eq. (2-6)) is found to be unsuitable for the global optimization algorithm, because of the complicated definition of the MAC value function (Eq. (2-7)). Therefore, in this chapter, the two global optimization algorithms will only be applied to eigenvector difference and modal dynamic residual formulations. In order to apply the B&B and P-RD algorithms to the two model updating formulations, the optimization problems shown in Eqs. (2-8) and (2-10) first requires reformulating. Application of the global optimization algorithms on the reformulated model updating formulations will be evaluated with both numerical simulation and laboratory experiment.

The rest of the chapter is organized as follows. The B&B algorithm is first introduced, followed by the introduction of the P-RD algorithm. Then, the reformulated eigenvector difference and modal dynamic residual formulations are presented. Finally, the reformulated model updating formulations with two global optimization algorithms will be evaluated with both numerical simulation and experimental data.

4.1 Branch-and-Bound Algorithm

The B&B algorithm is a global optimization algorithm that has the potential to solve the nonconvex optimization problems in FE model updating. The B&B algorithm involved in this research aims to solve a general non-convex optimization problem:

$$\underset{\mathbf{x}}{\text{minimize}}\ f(\mathbf{x}) \tag{4-1a}$$

$$\text{subject to}\ \ \boldsymbol{L}_{\mathbf{x}} \leq \mathbf{x} \leq \boldsymbol{U}_{\mathbf{x}} \tag{4-1b}$$

$$g_i(\mathbf{x}) \leq 0, i = 1 \ldots p \tag{4-1c}$$

$$h_i(\mathbf{x}) = 0, i = 1 \ldots q \tag{4-1d}$$

where $f: \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}$, $g_i: \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}, i = 1 \ldots p$ and $h_i: \mathbb{R}^{n_{\mathbf{x}}} \to \mathbb{R}, i = 1 \ldots q$ are scalar functions of vector variable $\mathbf{x} \in \mathbb{R}^{n_{\mathbf{x}}}$; In general, f and g_i can be nonconvex functions, and h_i may not be an affine function; $\boldsymbol{L}_{\mathbf{x}}$ and $\boldsymbol{U}_{\mathbf{x}}$ denote the lower and upper bounds of the optimization variable vector $\mathbf{x}$, respectively.

To illustrate the process of B&B algorithm, a low dimension non-convex objective function $f: \mathbb{R} \to \mathbb{R}$ with multiple local minimum is shown in Figure 4-1.

$$\underset{x}{\text{minimize}}\, f(x) \tag{4-2a}$$

$$\text{subject to } L_x \le x \le U_x \tag{4-2b}$$

At the first iteration, the B&B algorithm divides the box constraint of the optimization problem, $X = \{x | L_x \le x \le U_x\}$, into two non-overlapping subregions: $X = X_a^1 \cup X_b^1$ and $X_a^1 \cap X_b^1 = \emptyset$. X_a^1 and X_b^1 are the two non-overlapping subregions at the first iteration; the superscript represents the iteration number, and the subscript represents the region number. Instead of solving the original problem in Eq. (4-2) , two subproblems will be solved, one for each subregion.

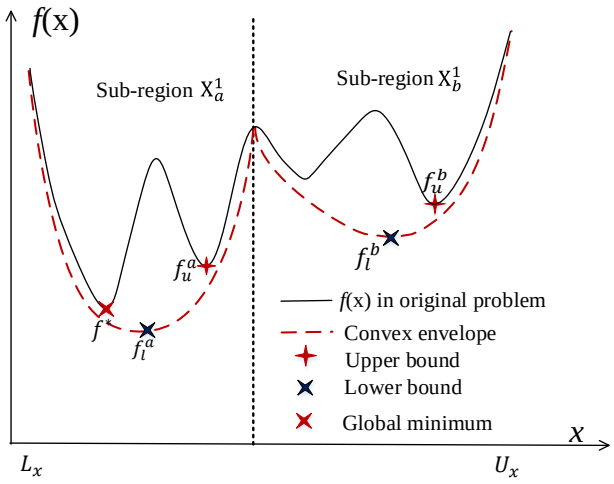


Figure 4-1. Illustration of convex envelope

When solving each subproblem, the B&B algorithm attempts to compute the lower and upper bounds of the optimal objective function value within each subregion. The upper bound of the optimal objective function value can be obtained using a local optimization algorithm (introduced in Section 2.4) upon each subproblem, denoted as f_u^a and f_u^b in Figure 4-1. A bigger challenge exists in computing the lower bound of the optimal

objective function value (f^*). Assuming f, g_i and h_i are all polynomial functions, within each subregion, this research adopts a convex-relaxation approach that constructs an underestimating convex envelope of the objective function [77]. Because of the underestimation, the optimal value of the relaxed convex problem is a lower bound of the objective function value for the subproblem. Figure 4-1 illustrates the convex envelopes, as well as the lower bounds (f_l^a and f_l^b) of the optimal objective function value within each subregion. Although the optimization problem shown in Eq. (4-2) only has an objective function $f(x)$ and a box constraint, convex envelope can also be constructed for the optimization problem shown in Eq. (4-1) which has nonconvex function $f(\mathbf{x})$, nonconvex inequality constraints $g_i(\mathbf{x})$, as well as non-affine equality constraints $h_i(\mathbf{x})$. Following equations show an example of convex relaxation for a bilinear term, $x_i x_j$, that could possibly exists in f, g_i or h_i.

$$x_i x_j \geq U_{x_i} x_j + x_i U_{x_j} - U_{x_i} U_{x_j} \tag{4-3a}$$

$$x_i x_j \geq L_{x_i} x_j + x_i L_{x_j} - L_{x_i} L_{x_j} \tag{4-3b}$$

$$x_i x_j \leq U_{x_i} x_j + x_i L_{x_j} - U_{x_i} L_{x_j} \tag{4-3c}$$

$$x_i x_j \leq L_{x_i} x_j + x_i U_{x_j} - L_{x_i} U_{x_j} \tag{4-3d}$$

Here the four formula at the right hand side of the inequality sign are affine with respect to x_i and x_j. When there is a bilinear term in the objective function or a constraint function, a new variable and a corresponding equality constraint will first be introduced, e.g. $y = x_i x_j$. Then, the equality constraint is relaxed to four inequalities demonstrated in Eq. (4-3). After relaxing all the bilinear terms in the optimization problem, within each

subregion, the original problem is relaxed to a convex one. Take the following problem as an example:

Prob. ①

$$\underset{\mathbf{x}}{\text{minimize}} \quad x_1x_2 + x_1x_3 \tag{4-4a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\mathbf{x}} \leq \mathbf{x} \leq \boldsymbol{U}_{\mathbf{x}} \tag{4-4b}$$

$$x_1x_2 \leq 0 \tag{4-4c}$$

$$x_2x_3 - 2 = 0 \tag{4-4d}$$

where the objective function $f(\mathbf{x}) = x_1x_2 + x_1x_3$ and the inequality constraint $g(\mathbf{x}) = x_1x_2$ are nonconvex; the equality constraint $h(\mathbf{x}) = x_2x_3 - 2$ is not affine. By observing three bilinear terms in the optimization problem, three new variables (y_1, y_2 and y_3) and three equality constraints are first added into Eq. (4-4). The optimization problem shown below is equivalent to Prob. ①.

Prob. ②

$$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad y_1 + y_2 \tag{4-5a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\mathbf{x}} \leq \mathbf{x} \leq \boldsymbol{U}_{\mathbf{x}} \tag{4-5b}$$

$$y_1 \leq 0 \tag{4-5c}$$

$$y_3 - 2 = 0 \tag{4-5d}$$

$$y_1 = x_1x_2 \tag{4-5e}$$

$$y_2 = x_1x_3 \tag{4-5f}$$

$$y_3 = x_2x_3 \tag{4-5g}$$

The nonconvex parts in Prob. ② are the equality constraints in Eqs. (4-5e) ~ (4-5g). To construct the convex envelope of Prob. ②, relaxation is introduced to the equality constrains in Eqs. (4-5e) ~ (4-5g).

Prob. ③

$$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad y_1 + y_2 \tag{4-6a}$$

$$\text{subject to} \quad \boldsymbol{L}_{\mathbf{x}} \leq \mathbf{x} \leq \boldsymbol{U}_{\mathbf{x}} \tag{4-6b}$$

$$y_1 \leq 0 \tag{4-6c}$$

$$y_3 - 2 = 0 \tag{4-6d}$$

$$y_1 \in \mathbf{Y}_1 \tag{4-6e}$$

$$y_2 \in \mathbf{Y}_2 \tag{4-6f}$$

$$y_3 \in \mathbf{Y}_3 \tag{4-6g}$$

where $\mathbf{Y}_i, i = 1,2,3$, are defined by four inequalities in Eq. (4-3). For example, $\mathbf{Y}_1$ is defined by the following four inequalities:

$$y_1 \geq U_{x_1} x_2 + x_1 U_{x_2} - U_{x_1} U_{x_2} \tag{4-7a}$$

$$y_1 \geq L_{x_1} x_2 + x_1 L_{x_2} - L_{x_1} L_{x_2} \tag{4-7b}$$

$$y_1 \leq U_{x_1} x_2 + x_2 L_{x_2} - U_{x_1} L_{x_2} \tag{4-7c}$$

$$y_1 \leq L_{x_1} x_2 + x_1 U_{x_2} - L_{x_1} U_{x_2} \tag{4-7d}$$

Because the four inequalities are affine on x_1 and x_2, the set $\mathbf{Y}_1$ in Eq. (4-7) is convex. Same conclusion can be made for $\mathbf{Y}_2$ and $\mathbf{Y}_3$. As a result, Prob. ③ is a convex

relaxation of Prob. ②, whose optimal value is a lower bound for the optimal value of Prob. ②. The detailed application of convex-relaxation approach on other form of polynomials can be found in [77].

After comparing the upper and lower bounds of the optimal value for the two subproblems in the first iteration, a new and tighter set of lower and upper bounds of the optimal objective function value, f_u^a and f_l^a in Figure 4-1, can be obtained. In the following iterations, the subregion with smaller lower bound value is further divided to update the upper and lower bounds of the optimal objective function value. The iteration continues until the gap between the upper and lower bounds are smaller than the prescribed tolerance. The global optimality of the B&B algorithm has been proved by previous researchers [41-43, 78]. To apply the B&B algorithm for FE model updating, a commercial optimization solver BARON is adopted [79].

4.2 Primal-Relaxed Dual Algorithm

The second global optimization algorithms to be studied in this research is the primal-relaxed dual (P-RD) algorithm [45, 46]. The P-RD algorithm is applicable for solving a particular type of nonconvex optimization problems, which is named as biconvex problems. A biconvex optimization problem has the following format:

Prob. ① $$\underset{\mathbf{x},\mathbf{y}}{\text{minimize}} \quad f(\mathbf{x},\mathbf{y}) \tag{4-8a}$$

$$\text{subject to } g_i(\mathbf{x},\mathbf{y}) \le 0, i = 1 \dots p \tag{4-8b}$$

$$h_i(\mathbf{x},\mathbf{y}) = 0, i = 1 \dots q \tag{4-8c}$$

Here the optimization variables are partitioned into two groups, i.e. $\mathbf{x} \in \mathbb{R}^{n_\mathbf{x}}$ and $\mathbf{y} \in \mathbb{R}^{n_\mathbf{y}}$. It is also required that $f(\mathbf{x},\mathbf{y})$, $g_i(\mathbf{x},\mathbf{y})$ and $h_i(\mathbf{x},\mathbf{y})$: $\mathbb{R}^{n_\mathbf{x}+n_\mathbf{y}} \to \mathbb{R}$ are continuous and piecewise differentiable. Please note that in the problem definition, the box constraints of $\mathbf{x}$ and $\mathbf{y}$ variable, i.e. $\boldsymbol{L}_\mathbf{x} \le \mathbf{x} \le \boldsymbol{U}_\mathbf{x}$ and $\boldsymbol{L}_\mathbf{y} \le \mathbf{y} \le \boldsymbol{U}_\mathbf{y}$, are included in $g_i(\mathbf{x},\mathbf{y})$. If the variable partitioning can satisfy the following requirements, the optimization problem is categorized as a biconvex problem:

- $f(\mathbf{x},\mathbf{y})$ is convex in $\mathbf{x}$ for every fixed $\mathbf{y}$, and vice versa;
- $g(\mathbf{x},\mathbf{y})$ is convex in $\mathbf{x}$ for every fixed $\mathbf{y}$, and vice versa;
- $h(\mathbf{x},\mathbf{y})$ is affine in $\mathbf{x}$ for every fixed $\mathbf{y}$, and vice versa.

4.2.1 Introduction of P-RD Algorithm

Benefiting from the biconvex characteristics of the problem, the P-RD algorithm iteratively approximates and improves the lower bound of the optimal value by solving relaxed dual subproblems. In the meantime, at each iteration, the upper bound of the optimal value is updated by solving Prob. ① with $\mathbf{y}$ variables fixed at selected feasible values. When the gap between the lower and upper bound is smaller than a prescribed tolerance value, the P-RD algorithm decides that the optimization process reaches the global optimal value of the optimization problem. Before introducing the P-RD algorithm, the Lagrangian function, $L(\mathbf{x},\mathbf{y},\boldsymbol{\lambda},\boldsymbol{\nu})$: $\mathbb{R}^{n_\mathbf{x}+n_\mathbf{y}+p+q} \to \mathbb{R}$, of Prob. ① is defined [36]:

$$L(\mathbf{x},\mathbf{y},\boldsymbol{\lambda},\boldsymbol{\nu}) = f(\mathbf{x},\mathbf{y}) + \sum_{i=1}^{p} \lambda_i\, g_i(\mathbf{x},\mathbf{y}) + \sum_{i=1}^{q} \nu_i\, h_i(\mathbf{x},\mathbf{y}),\ \lambda_i \ge 0 \tag{4-9}$$

where λ_i and ν_i are the Lagrange multipliers associated with the equality and inequality constraints in Prob. ①, respectively. In addition, the Lagrange dual function $G(\boldsymbol{\lambda}, \boldsymbol{\nu})$: $\mathbb{R}^{p+q} \rightarrow \mathbb{R}$ is defined as the infimum value of the Lagrangian function over $\mathbf{x}$ and $\mathbf{y}$.

$$G(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x},\mathbf{y}} L(\mathbf{x}, \mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\mathbf{x},\mathbf{y}} \left\{ f(\mathbf{x}, \mathbf{y}) + \sum_{i=1}^{p} \lambda_i \, g_i(\mathbf{x}, \mathbf{y}) + \sum_{i=1}^{q} \nu_i \, h_i(\mathbf{x}, \mathbf{y}) \right\} \tag{4-10}$$

Because $G(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is the pointwise infimum of affine functions of $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$, it is well known to be concave. The function value is a lower bound to the optimal value of Eq. (4-8) for any $\lambda_i \geq 0$ and any ν_i (recall that $g_i(\mathbf{x}, \mathbf{y}) \leq 0$). The lower bound feature of the Lagrange dual function is the basis of P-RD algorithm.

The P-RD algorithm computes the lower and upper bounds of the optimal objective function value, and iteratively minimizes the gap between the bounds. Owing to the special structure of the biconvex problem, the lower bound of the optimal value is computed by solving a series of relaxed dual subproblems. To illustrate the process of P-RD algorithm, we consider a low dimension function $f(x, y)$: $\mathbb{R}^2 \rightarrow \mathbb{R}$ with scalar variables. At the beginning, Prob. ① is projected onto the space of variable *y*, shown as black solid line in Figure 4-2 [80]. $v(y)$ in the figure is defined in Eq. (4-11), where a feasible set of variable y, is defined as $\mathbf{V} = \{y: g_i(x, y) \leq 0, h_i(x, y) = 0 \text{ for some } x: L_x \leq x \leq U_x\}$. Likewise, the box constraint of variable y can also be defined as $\mathbf{Y} = \{y: L_y \leq y \leq U_y\}$.

$$v(y) = \inf_{x} \left\{ f(x, y) \middle| \begin{matrix} g_i(x, y) \leq 0, i = 1 \ldots p \\ h_i(x, y) = 0, i = 1 \ldots q \end{matrix} \right\} \tag{4-11a}$$

$$y \in \mathbf{Y} \cap \mathbf{V} \tag{4-11b}$$

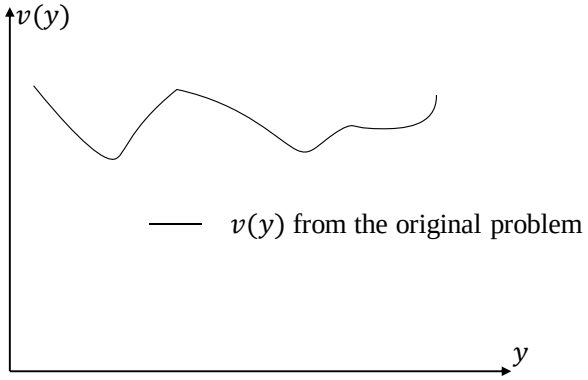


Figure 4-2. Projection of Prob. ① onto the space of variable y

The infimum in Eq. (4-11a) is parametric on a fixed value of y, so $v(y)$ can be treated as a set of solutions of Prob. ① solved for different fixed values of y. As a result, Prob. ① and ② are considered to be equivalent.

Prob. ② $$\underset{y}{\text{minimize}} \quad v(y) \tag{4-12}$$

Also, it is worth noting that $v(y)$ in Prob. ② is nonconvex, because Prob. ① is nonconvex. Furthermore, because the Lagrange dual function yields a lower bound of the primal problem, Prob. ③ shown below is equivalent to Prob. ②.

Prob. ③ $$\underset{\mu_{\mathrm{B}},y,\boldsymbol{\lambda},\boldsymbol{\nu}}{\text{minimize}} \quad \mu_{\mathrm{B}} \tag{4-13a}$$

$$\text{subject to} \quad \mu_{\mathrm{B}} \geq \inf_{x} \{L(x,y,\boldsymbol{\lambda},\boldsymbol{\nu}) | L_x \leq x \leq U_x\} \tag{4-13b}$$

$$y \in \mathbf{Y} \cap \mathbf{V} \tag{4-13c}$$

In addition to x and y in Prob. ②, Lagrangian multiplies ($\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$) and a scalar variable (μ_{B}), are added into the optimization variables. Prob. ③ aims to find the infimum value (μ_{B}) of the Lagrangian function. However, Prob. ③ is as difficult as Prob. ① for the following two reasons. First, the constraints in Eq. (4-13c) involve **V**, an implicit and non-convex set of *y*, which makes the problem hard to solve; Second, the problem is still parametric in x. To overcome the difficulty, the P-RD algorithm drops the constraint $y \in \mathbf{Y} \cap \mathbf{V}$ in Prob. ③, which results in the relaxed dual subproblem. In the 1st iteration, the P-RD algorithm starts from y^0, an initial value for variable *y*. With a fixed y^0**,** the convex Prob. ④ is solved, as shown in Figure 4-3.

Prob. ④

$$\underset{x}{\text{minimize}}\, f(x, y^0) \tag{4-14a}$$

$$\text{subject to } g_i(x, y^0) \le 0, i = 1 \ldots p \tag{4-14b}$$

$$h_i(x, y^0) = 0, i = 1 \ldots q \tag{4-14c}$$

$$L_x \le x \le U_x \tag{4-14d}$$

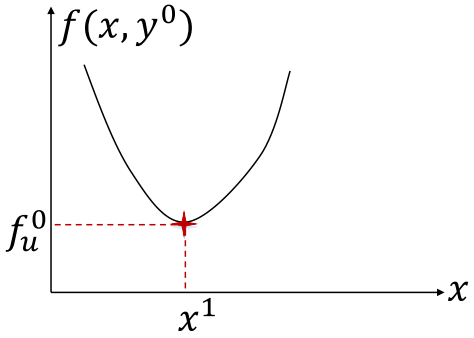


Figure 4-3. Illustration of solving Prob. ④

A feasible solution of Prob. ④ provides an updated upper bound of the optimal objective function value for Prob. ① (f_u^0), the corresponding optimal point (x^1), as well as the Lagrange multipliers for inequality and equality constraints ($\boldsymbol{\lambda}^1$ and $\boldsymbol{\nu}^1$). The Lagrangian function of Prob. ④ can be formulated as follows

$$L^1\left(x, y, \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1\right) = f(x,y) + \sum_{i=1}^{p} \lambda_i^1 g_i(x,y) + \sum_{i=1}^{q} \nu_i^1 h_i(x,y) \tag{4-15}$$

Figure 4-4 shows the solution of Prob. ④ in the space of variable *y*, which also corresponds to the initial condition of the P-RD algorithm. Some notations are introduced first in order to continue the P-RD algorithm. We define **CB** to represent all possible combinations of the lower and upper bounds of variable *x*. Taking $\mathbf{x} \in \mathbb{R}^2$ as an example, if the bounds are denoted as $\begin{Bmatrix} L_{x_1} \\ L_{x_2} \end{Bmatrix}$ and $\begin{Bmatrix} U_{x_1} \\ U_{x_2} \end{Bmatrix}$, combination of bounds includes $\mathbf{CB} = \left\{\{L_{x_1}, L_{x_2}\}, \{L_{x_1}, U_{x_2}\}, \{U_{x_1}, L_{x_2}\}, \{U_{x_1}, L_{x_2}\}\right\}$.

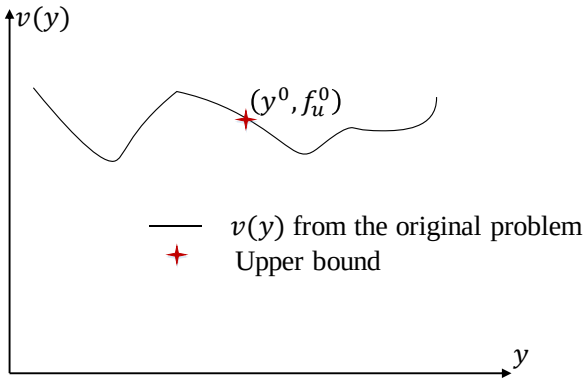


Figure 4-4. Illustration of P-RD algorithm (Initial condition)

In this low dimensional illustration, **CB** contains only two elements, $\{\{U_x\}, \{L_x\}\}$. Additionally, $B^l, l = 1,2$, denotes an element in **CB**, i.e. $B^l = U_x$ or L_x. Recall x^1 is the

solution of Prob. ④. With $\mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}$ representing the partial derivative of Eq. (4-15) with respect to x and evaluated at x^1, $L^1(x,\mathrm{y};\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}} = L^1(x^1,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1) + \mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}\cdot(x-x^1)$ denotes the linearization of the Lagrangian function with respect to variable x at x^1. Owing to the convexity of Prob. ④, $L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$ is always smaller than $L(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)$ for any $\{x|L_x \le x \le U_x\}$. Additionally, because $L^1(x,\mathrm{y};\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$ is affine on x, its minimum value must take place when the value of x hits a bound, U_x or L_x. Which bound gives the minimum value depends on the sign of the derivative, $\mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}$. In other words, if $\mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1} \le 0$, $L^1(x,\mathrm{y};\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$ achieves minimum at $x = U_x$. On the contrary, if $\mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1} \ge 0$, $L^1(x,\mathrm{y};\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$ achieves minimum at $x = L_x$. The relaxed dual subproblem is formulated as follows.

Prob. ⑤
$$\underset{y,\mu_{\mathrm{B}}}{\text{minimize}}\ \mu_{\mathrm{B}} \tag{4-16a}$$

$$\text{subject to } \mu_{\mathrm{B}} \ge L^1(B^l,\mathrm{y};\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}} \tag{4-16b}$$

$$\begin{cases} \mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1} \le 0, & \text{if } B^l = U_x \\ \mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1} \ge 0, & \text{if } B^l = L_x \end{cases}, l = 1,2 \tag{4-16c}$$

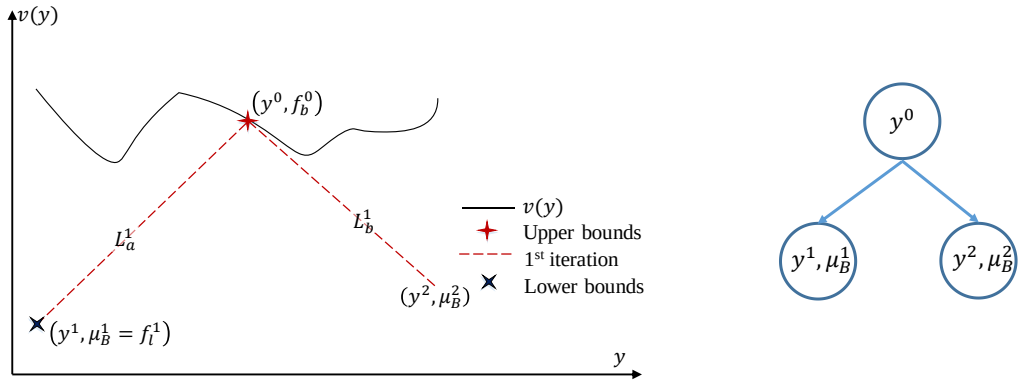
$$L_y \le y \le U_y \tag{4-16d}$$

Here μ_{B} and y are the optimization variables. Prob. ⑤ aims to find the minimum value (μ_{B}) of the linearized Lagrangian function over the y variable with the accompanying derivative constraint, i.e. the sign of $\mathrm{D}_x L^1(x,y;\boldsymbol{\lambda}^1,\boldsymbol{\nu}^1)|_{x^1}$. The P-RD algorithm requires Prob. ⑤ to be solved using all the elements in **CB**, i.e. $B^l \in \{\{U_x\},\{L_x\}\}, l = 1,2$ for this

example. The 1st iteration generates two relaxed dual subproblems, denoted as L_a^1 and L_b^1 in Figure 4-5(a). Without loss of generality, we define that corresponding to $B^l = U_x$, $L_a^1 = L^1(U_x, \mathrm{y}; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$, so the accompanying derivative constraint is $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1} \leq 0$. Corresponding to $B^l = L_x$, L_b^1 is equal to $L^1(L_x, \mathrm{y}; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$, whose accompanying derivative constraint is $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1} \geq 0$. The constraint shown in Eq. (4-16c) represents the feasible range of the relaxed dual subproblem. As illustrated in Figure 4-5(a), $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1} \leq 0$ is equivalent to $L_y \leq y \leq y^0$. Likewise, $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1} \geq 0$ is equivalent to $y^0 \leq y \leq U_y$. Because Prob. ① is biconvex, the relaxed dual subproblem (Prob. ⑤) is convex on μ_{B} and y. Compared to Prob. ③, Prob. ⑤ first drops the constraints of the feasible set $\mathbf{V}$ in Eq. (4-13c). Prob. ⑤ also replaces the Lagrangian function constraint and the optimization variable x in Eq. (4-13b) with the linearized Lagrangian function and the combinations of lower and upper bounds, i.e. B^l from $\mathbf{CB}$, respectively. Therefore, Prob. ⑤ provides an underestimating envelope of Prob. ③. In other words, the dashed lines (L_a^1 and L_b^1) in Figure 4-5(a) are always smaller than or equal to the black solid line ($v(y)$) when the dashed lines are within the range of the accompanying derivative constraint.

To implement the following iterations of the P-RD algorithm, a branch-and-bound (B&B) framework is adopted [47] (Figure 4-5(b)). The root node of the B&B framework stores the initial value for y, i.e. y^0. For this illustration example, after solving two relaxed dual subproblems for the 1st iteration, two new "child" nodes are generated from the root node. Within each "child" node, following information is stored: (i) optimal value of y and μ_{B} from Prob. ⑤; (ii) the constraint of the linearized Lagrangian function in Eq. (4-16b)

of Prob. ⑤; (iii) accompanying derivative constraint, i.e. the sign of $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1}$ in Eq. (4-16c) of Prob. ⑤. For example, the constraint of the linearized Lagrangian function stored in node (y^1, μ_B^1) is $\mu_{\mathrm{B}} \geq L^1(U_x, \mathrm{y}; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}}$, and the stored accompanying derivative constraint is $\mathrm{D}_x L^1(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1} \leq 0$.



(a) Illustration of relaxed dual subproblems (b) Illustration of B&B framework

Figure 4-5. Illustration of P-RD algorithm (1st iteration)

After solving two relaxed dual subproblems of the 1st iteration, solutions of both subproblems, i.e. μ_B^1 and μ_B^2 will be stored in a lower bound set, $\boldsymbol{\mu}_B$, to determine the lower bound of the objective function value, i.e. $\boldsymbol{\mu}_B = \{\mu_B^1, \mu_B^2\}$. At this point, the "child" node with the smallest value within $\boldsymbol{\mu}_B$ is selected as the "parent" node for the next iteration. It can be seen from Figure 4-5(a) that $\mu_B^1 < \mu_B^2$ in this example. Thus, μ_B^1 will be set as the lower bound of the objective function value, i.e. $f_l^1 = \mu_B^1$, and the node (y^1, μ_B^1) becomes the "parent" node for next iteration. In the meantime, μ_B^1 is deleted from the lower bound set $\boldsymbol{\mu}_B$. Otherwise, μ_B^1 will always be the minimum value in $\boldsymbol{\mu}_B$, and the lower bound of the objective function value cannot be updated through the following iterations. In other words, the lower bound set after the 1st iteration only has one component, i.e. $\boldsymbol{\mu}_B = \{\mu_B^2\}$.

At the 2nd iteration, Prob. ④ is solved by changing y^0 to y^1. The optimal solution point is named as x^2. The optimal objective function value of Prob. ④ is named as f_u^1, since the value provides an updated upper bound of the optimal objective function value of Prob. ① (Figure 4-6(a)). Meanwhile, the corresponding Lagrange multipliers, $\boldsymbol{\lambda}^2$ and $\boldsymbol{\nu}^2$ are obtained. Consequently, Prob. ⑥ is the relaxed dual subproblem to be solved at the 2nd iteration, where the constraint of the linearized Lagrangian function and accompanying derivative constraint for the current iteration are first added into the constraint set (Eqs. (4-17b) and (4-17c)), shown as L_a^2 in Figure 4-6(a). $L^2(B^l, \mathrm{y}; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2)|_{x^2}^{\mathrm{lin}}$ in Eq. (4-17b) equals $L^2(x^2, y; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2) + \mathrm{D}_x L^2(x, y; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2)|_{x^2} \cdot (B^l - x^1)$. At the 2nd iteration, only one relaxed dual subproblem will be solved, because the fixed value of y is taken at its bound, i.e. $y^1 = L_x$.

Prob. ⑥

$$\underset{y,\mu_{\mathrm{B}}}{\text{minimize}} \quad \mu_{\mathrm{B}} \tag{4-17a}$$

$$\text{subject to} \quad \mu_{\mathrm{B}} \geq L^2(B^l, \mathrm{y}; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2)|_{x^2}^{\mathrm{lin}} \tag{4-17b}$$

$$\begin{cases} \mathrm{D}_x L^2(x, y; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2)|_{x^2} \leq 0, & \text{if } B^l = U_x \\ \mathrm{D}_x L^2(x, y; \boldsymbol{\lambda}^2, \boldsymbol{\nu}^2)|_{x^2} \geq 0, & \text{if } B^l = L_x \end{cases} \tag{4-17c}$$

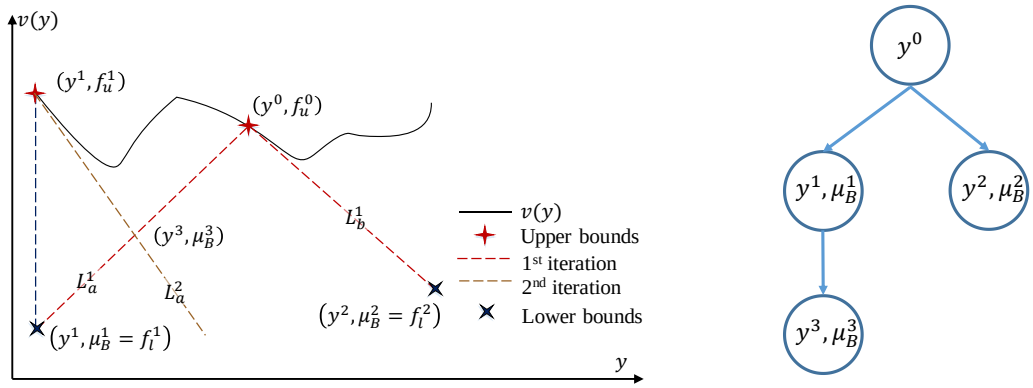
$$\mu_{\mathrm{B}} \geq L_a^1(U_x, \mathrm{y}; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1)|_{x^1}^{\mathrm{lin}} \tag{4-17d}$$

$$\mathrm{D}_x L^1\left(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1\right)|_{x^1} \leq 0 \tag{4-17e}$$

$$L_y \leq y \leq U_y \tag{4-17f}$$

In addition, one constraint of the linearized Lagrangian function and the accompanying derivative constraint from its "parent" node, i.e. the node (y^1, μ_B^1) in the B&B framework, are added into the constraint set (Eqs. (4-17d) and (4-17e)), shown as L_a^1 in Figure 4-6(a). Similar to the 1st iteration, the combination of constraints shown in Eqs.

(4-17b) and (4-17e) represent the feasible range of the relaxed dual subproblem, i.e. $L_y \leq y \leq y^0$ in Figure 4-6(a). Figure 4-6(b) displays the B&B framework after the 2nd iteration, where one new 'child' node (y^3, μ_B^3) is added. Accordingly, μ_B^3 is added into the lower bound set, i.e. $\boldsymbol{\mu}_B = \{\mu_B^2, \mu_B^3\}$.



(a) Illustration of relaxed dual subproblems (b) Illustration of B&B framework

Figure 4-6. Illustration of P-RD algorithm (2nd iteration)

At the end of 2nd iteration, the smallest value in $\boldsymbol{\mu}_B$ becomes the updated lower bound of the objective function value. In this illustration, it can be seen that μ_B^2 is smaller than μ_B^3, so μ_B^2 is set as f_l^2, and node (y^2, μ_B^2) becomes the "parent" node for the next iteration. Also, μ_B^2 will be deleted from $\boldsymbol{\mu}_B$ for future comparison, which means that the lower bound set after 2nd iteration becomes $\boldsymbol{\mu}_B = \{\mu_B^3\}$.

4.2.2 Pseudo Code for the P-RD Algorithm

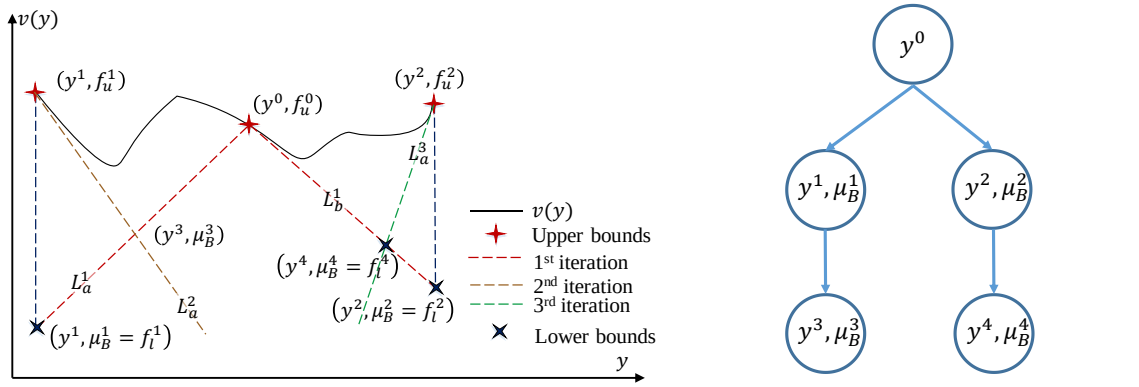
For each later iteration, the P-RD algorithm requires to add one constraint of the linearized Lagrangian function and the accompanying derivative constraints from each "parent" node in the B&B framework to the constraint set of the relaxed dual subproblem

for the new "child" node. The pseudo code below first shows how to correctly construct a constraint set, $\mathbf{R}_c$, from each "parent" node in the B&B framework.

```
Initialize R_c = ∅, and set N_m as the "parent" node of the new
"child" node.
while(N_m is not the root node in the B&B framework) {
     i) Add the constraint of linearized Lagrangian function
        and the accompanying derivative constraints stored in
        Node N_m into R_c.
     ii) Change N_m to the "parent" node of N_m; move up one
         level
 }
Add R_c into the constraint set of relaxed dual subproblem for
the new "child" node.
```

Figure 4-7. Pesudo code of constructing $\mathbf{R}_c$

The pseudo code can be explained with the 3rd iteration of the illustration example shown above. Prob. ④ is first solved by changing y to y^2, from which an updated upper bound of the optimal objective function value, shown as f_u^2 in Figure 4-8(a), updated values for x variable, x^3, as well as Lagrange multipliers ($\boldsymbol{\lambda}^3$ and $\boldsymbol{\nu}^3$) can be obtained.



(a) Illustration of relaxed dual subproblems (b) Illustration of B&B framework

Figure 4-8. Illustration of P-RD algorithm (3rd iteration)

When constructing the constraint set, $\mathbf{R}_{\mathrm{c}}$, shown as Step 1 in the pseudo code, N_m is set as node (y^2, μ_B^2) ("parent" node of the new "child" node). After the first "`while`" loop, the constraint set, $\mathbf{R}_{\mathrm{c}}$, becomes as follows:

$$\mathbf{R}_{\mathrm{c}} = \left\{ y, \mu_{\mathrm{B}} \middle| \begin{matrix} \mu_{\mathrm{B}} \geq L_b^1 \left(L_x, \mathrm{y}; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1 \right) |_{x^1}^{\mathrm{lin}} \\ \mathrm{D}_x L^1 \left(x, y; \boldsymbol{\lambda}^1, \boldsymbol{\nu}^1 \right) |_{x^1} \geq 0 \end{matrix} \right\} \tag{4-18}$$

According to the pseudo code, N_m is then set as node y^0, i.e. the "parent" node of (y^2, μ_B^2) . Now that node y^0 is the root node in the B&B framework, the pseudo code will exit the "`while`" loop and $\mathbf{R}_{\mathrm{c}}$ is added into the constraint set of Prob. ⑦. As a result, the relaxed dual subproblem of the 3rd iteration can be formulated. Similar to the 2nd iteration, because the fixed value of y is taken at its bound, i.e. $y^2 = U_x$, only one relaxed dual subproblem is constructed and solved at the 3rd iteration.

Prob. ⑦

$$\underset{y, \mu_{\mathrm{B}}}{\text{minimize}} \quad \mu_{\mathrm{B}} \tag{4-19a}$$

$$\text{subject to} \quad \mu_{\mathrm{B}} \geq L^3(B^l, \mathrm{y}; \boldsymbol{\lambda}^3, \boldsymbol{\nu}^3)|_{x^3}^{\mathrm{lin}} \tag{4-19b}$$

$$\begin{cases} \mathrm{D}_x L^3(x, y; \boldsymbol{\lambda}^3, \boldsymbol{\nu}^3)|_{x^3} \leq 0, & \text{if } B^l = U_x \\ \mathrm{D}_x L^3(x, y; \boldsymbol{\lambda}^3, \boldsymbol{\nu}^3)|_{x^3} \geq 0, & \text{if } B^l = L_x \end{cases} \tag{4-19c}$$

$$\{y, \mu_{\mathrm{B}}\} \in \mathbf{R}_{\mathrm{c}} \tag{4-19d}$$

$$L_y \leq y \leq U_y \tag{4-19e}$$

After solving the relaxed dual subproblem of the current iteration, the B&B framework is shown as Figure 4-8(b). Also, the lower bound set is updated as $\boldsymbol{\mu}_B = \{\mu_B^3, \mu_B^4\}$.

The pseudo code shown in Figure 4-9 illustrates the process of P-RD algorithm. We first define a generalized form of the primal problem:

$$\text{Prob. ⑧} \qquad \underset{\mathbf{x}}{\text{minimize}} \;\; f(\mathbf{x}, \mathbf{y}^K) \tag{4-20a}$$

$$\text{subject to } g_i(\mathbf{x}, \mathbf{y}^K) \leq 0, i = 1 \ldots p \tag{4-20b}$$

$$h_i(\mathbf{x}, \mathbf{y}^K) = 0, i = 1 \ldots q \tag{4-20c}$$

Similarly, a generalized form of the relaxed-dual subproblem can be defined as below:

$$\text{Prob. ⑨} \qquad \underset{\mathbf{y}, \mu_\mathrm{B}}{\text{minimize}} \;\; \mu_\mathrm{B} \tag{4-21a}$$

$$\text{subject to } \; \{\mathbf{y}, \mu_\mathrm{B}\} \in \mathbf{R}_\mathrm{c} \cap \mathbf{R}_{\mathrm{c1}} \tag{4-21b}$$

$$\boldsymbol{L}_\mathbf{y} \leq \mathbf{y} \leq \boldsymbol{U}_\mathbf{y} \tag{4-21c}$$

where $\mathbf{R}_\mathrm{c}$ is the constraints from the nodes in the higher level of the B&B framework; $\mathbf{R}_{\mathrm{c1}}$ is the constraints from the current iteration. The P-RD algorithm first assigns the tolerance value between the lower and upper bound of the optimal objective function value (ε) and the maximum iteration number ($K_{\max}$). The P-RD algorithm also selects initial values for $\mathbf{y}$ variables ($\mathbf{y}^0$) and solves Prob. ⑧. Then, the constraint set $\mathbf{R}_\mathrm{c}$ is constructed based on the pseudo code shown in Figure 4-7 and added into the relaxed dual subproblem, Prob. ⑨. The next step is to add the constraint of the linearized Lagrangian function and the accompanying derivative constraints for the current iteration into the relaxed dual subproblem, denoted as $\mathbf{R}_{\mathrm{c1}}$. By solving the relaxed dual subproblem (Prob. ⑨), a lower bound of the optimal objective function value (μ_B^i) and the corresponding values of $\mathbf{y}$

variables ($\mathbf{y}_i$) can be obtained. At each iteration of P-RD algorithm, Prob. ⑨ is constructed with different $\boldsymbol{B}^i \in \mathbf{CB}$, and in total to be solved by $2^{n_\mathbf{x}}$ times. At the end of iteration, an updated lower bound of the optimal objective function value is selected from $\mu_\mathbf{B}$ and the corresponding values of $\mathbf{y}$ variables are used to solve Prob. ⑧ for the next iteration. After solving Prob. ⑧, the updated values for $\mathbf{x}$ and $\mathbf{y}$ variables, i.e. $\mathbf{x}^{K+1}$ and $\mathbf{y}^{K+1}$, are used as initial values for the local search algorithm to solve Prob. ① to get better f_u^{K+1}. The algorithm will continue when either the gap between the lower and upper bound value is larger than tolerance, i.e. $|f_u^K - f_l^K| > \varepsilon$, or the iteration number is smaller than the maximum iteration number, i.e. $K < K_{\max}$. Detailed description on the derivation and global optimality proof of P-RD algorithm can be found in [47].

Set tolerance between lower and upper bound of the optimal objective function value as ε;
Set the maximum iteration number as $K_{\max}$;
Initialize $K = 0$, $\mathbf{y}^0$, $\mu_\mathbf{B} = \emptyset$;
Solve Prob. ⑧ to get $\{f_u^0, \mathbf{x}^0, \boldsymbol{\lambda}^0, \boldsymbol{\nu}^0\}$;
while ($|f_u^K - f_l^K| > \varepsilon$ and $K < K_{\max}$) {
  Construct $\mathbf{R}_\mathbf{c}$ with Figure 4-7
  for $i = 1, 2, 3, \ldots, 2^{n_\mathbf{x}}$ {
    Select $\boldsymbol{B}^i \in \mathbf{CB}$;
    Add $\mu_\mathrm{B} \geq L^K(\boldsymbol{B}^i, \mathbf{y}; \boldsymbol{\lambda}^K, \boldsymbol{\nu}^K)|^{\mathrm{lin}}_{\mathbf{x}^K}$ into $\mathbf{R}_{\mathbf{c}1}$;
    for $j = 1, 2, 3, \ldots, n_\mathbf{x}$ {
      if $B_j^i = U_{x_j}$
        Add $\mathrm{D}_{x_j} L^K(\mathbf{x}, \mathbf{y}; \boldsymbol{\lambda}^K, \boldsymbol{\nu}^K)\big|_{\mathbf{x}^K} \leq 0$ into $\mathbf{R}_{\mathbf{c}1}$;
      else
        Add $\mathrm{D}_{x_j} L^K(\mathbf{x}, \mathbf{y}; \boldsymbol{\lambda}^K, \boldsymbol{\nu}^K)\big|_{\mathbf{x}^K} \geq 0$ into $\mathbf{R}_{\mathbf{c}1}$;
    }

```
    Solve Prob. ⑨ to get {μ_B^i, y_i};
    Add μ_B^i into μ_B;
  }
  Find minimum value in μ_B and set the minimum value as f_l^{K+1};
  Set the corresponding y as y^{K+1};
  Remove f_l^{K+1} from μ_B;
  Solve Prob. ⑧ to get {f_u^{K+1}, x^{K+1}, μ^{K+1}, λ^{K+1}};
  Set {x^{K+1}, y^{K+1}} as initial value and use local search
  optimization algorithm to solve Prob. ① to get better f_u^{K+1}
  K = K + 1;
}
```

Figure 4-9. Pseudo code of P-RD algorithm

4.3 FE Model Updating Formulations for Global Optimization Algorithms

To apply the global optimization algorithms to the finite element (FE) model updating problems, the model updating formulations introduced in CHAPTER 2 requires reformulating. This subsection will present the eigenvector difference and modal dynamic residual formulations solvable by the B&B and P-RD algorithms.

4.3.1 Eigenvector Difference Formulation

As described in Section 2.2.2, the objective function of the eigenvector difference formulation, Eq. (2-8), is an oracle formulation of the updating variable $\boldsymbol{\alpha}$, and thus is not applicable to either B&B or P-RD algorithm. Prior to the reformulation, we recall the relevant notation. The stiffness matrix $\mathbf{K}(\boldsymbol{\alpha})$ equals $\mathbf{K_0} + \sum_{j=1}^{n_\alpha} \alpha_j \mathbf{K}_j$, where $\boldsymbol{\alpha} \in \mathbb{R}^{n_\alpha}$ is a vector corresponds to the stiffness parameters to be updated. The mass matrix is denoted as $\mathbf{M}$ and assumed to be accurate enough. λ_i^{EXP} and λ_i are the experimental and simulated eigenvalue. $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}$ stand for the experimental and simulated eigenvector at

measured degrees of freedom (DOF), which are both normalized so that the entry corresponding to the largest magnitude (the q_i-th entry) equals 1. $\boldsymbol{\psi}_i^{\mathrm{u}}$ is the simulated eigenvector at unmeasured DOFs. The eigenvector difference formulation in Eq. (2-8) is reformulated as follows:

$$\underset{\boldsymbol{\alpha},\boldsymbol{\Lambda},\boldsymbol{\varphi}_{-q}^{m},\boldsymbol{\varphi}^{\mathrm{u}},\boldsymbol{\delta}}{\text{minimize}} \quad \|\boldsymbol{\delta}\|_1 \tag{4-22a}$$

$$\text{subject to } -\boldsymbol{\delta}_i \leq \begin{bmatrix} \frac{\lambda_i^{\mathrm{EXP}}-\lambda_i}{\lambda_i^{\mathrm{EXP}}} \cdot w_{\lambda_i} \\ \left\{\boldsymbol{\psi}_{-q_i,i}^{\mathrm{EXP,m}} - \boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}\right\} \cdot w_{\boldsymbol{\psi}_i} \\ w_{\mathrm{eig}} \cdot [\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i \mathbf{M}] \begin{Bmatrix} \boldsymbol{\psi}_i^{\mathrm{m}} \\ \boldsymbol{\psi}_i^{\mathrm{u}} \end{Bmatrix} \end{bmatrix} \leq \boldsymbol{\delta}_i, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-22b}$$

$$\boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{4-22c}$$

$$\boldsymbol{L}_{\lambda_i} \leq \lambda_i \leq \boldsymbol{U}_{\lambda_i}, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-22d}$$

$$\boldsymbol{L}_{\boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}} \leq \boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}} \leq \boldsymbol{U}_{\boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}}, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-22e}$$

$$\boldsymbol{L}_{\boldsymbol{\psi}_i^{\mathrm{u}}} \leq \boldsymbol{\psi}_i^{\mathrm{u}} \leq \boldsymbol{U}_{\boldsymbol{\psi}_i^{\mathrm{u}}}, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-22f}$$

where $\|\cdot\|_1$ represents the $\mathcal{L}_1$ norm of a vector; the optimization variables include $\boldsymbol{\alpha} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$; simulated eigenvalues $\boldsymbol{\Lambda} = [\lambda_1 \quad \lambda_2 \quad \cdots \quad \lambda_{n_{\mathrm{modes}}}] \in \mathbb{R}^{n_{\mathrm{modes}}}$; the simulated eigenvectors at measured DOFs except for the q_i -th entry $\boldsymbol{\varphi}_{-q}^{\mathrm{m}} = \left[\boldsymbol{\psi}_{-q_1,1}^{\mathrm{m}} \quad \boldsymbol{\psi}_{-q_2,2}^{\mathrm{m}} \quad \cdots \quad \boldsymbol{\psi}_{-q_{n_{\mathrm{modes}}},n_{\mathrm{modes}}}^{\mathrm{m}}\right]^{\mathrm{T}} \in \mathbb{R}^{(n_{\mathrm{m}}-1)\cdot n_{\mathrm{modes}}}$; the simulated eigenvectors at unmeasured DOFs $\boldsymbol{\varphi}^{\mathrm{u}} = [\boldsymbol{\psi}_1^{\mathrm{u}} \quad \boldsymbol{\psi}_2^{\mathrm{u}} \quad \cdots \quad \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{u}}]^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{u}}\cdot n_{\mathrm{modes}}}$, as well as a vector variable $\boldsymbol{\delta} \in \mathbb{R}^{(n_{\mathrm{m}}+N)\cdot n_{\mathrm{modes}}}$. The vector variable $\boldsymbol{\delta}$ equals $[\boldsymbol{\delta}_1 \quad \boldsymbol{\delta}_2 \quad \cdots \quad \boldsymbol{\delta}_{n_{\mathrm{modes}}}]^{\mathrm{T}}$, where $\boldsymbol{\delta}_i \in \mathbb{R}^{(n_{\mathrm{m}}+N)}$. In total, the number of optimization variables equal $n_{\boldsymbol{\alpha}} + n_{\mathrm{modes}} \cdot$

$N + (n_{\mathrm{m}} + N) \cdot n_{\mathrm{modes}}$; w_{λ_i} and $w_{\boldsymbol{\psi}_i}$ represents the weighting factor of eigenvalue and eigenvector difference for the i-th mode; w_{eig} denotes the weighting factor of the generalized eigenvalue equation; The sign "$\preceq$" in Eq. (4-22b) is overloaded to represent element-wise inequality; Eqs. (4-22c) ~ (4-22f) represent the lower and bounds of the optimization variables. In CHAPTER 2, it is assumed that the experimental eigenvector is normalized so that the q_i-th entry equals 1, so accordingly, $\boldsymbol{\psi}_i^{\mathrm{m}}$ also has 1 at the q_i-th entry and $\boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}$ at the rest of entries. Recall the definition of matrix $\mathbf{Q}_i$ in Eq. (2-9), $\left\{\boldsymbol{\psi}_{-q_i,i}^{\mathrm{EXP,m}} - \boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}\right\}$ in Eq. (4-22b) equals $\mathbf{Q}_i\left\{\boldsymbol{\psi}_i^{\mathrm{EXP,m}} - \boldsymbol{\psi}_i^{\mathrm{m}}(\boldsymbol{\alpha})\right\}$ in Eq. (2-8). The residual vector $\mathbf{r}_i(\boldsymbol{\alpha})$ defined in Eq. (2-25) is equivalent to $\begin{bmatrix} \frac{\lambda_i^{\mathrm{EXP}} - \lambda_i}{\lambda_i^{\mathrm{EXP}}} \cdot w_{\lambda_i} \\ \left\{\boldsymbol{\psi}_{-q_i,i}^{\mathrm{EXP,m}} - \boldsymbol{\psi}_{-q_i,i}^{\mathrm{m}}\right\} \cdot w_{\boldsymbol{\psi}_i} \end{bmatrix}$ in Eq. (4-22b). In addition, Eq. (2-8) solves the generalized eigenvalue equation in Eq. (2-5) with updated value of $\boldsymbol{\alpha}$ to obtain the simulated eigenvalue and eigenvectors, λ_i and $\boldsymbol{\psi}_i$, while Eq. (4-22b) adds the generalized eigenvalue equation explicitly into the constraints of the optimization problem with a weighting factor equal to w_{eig}. As a result, Eq. (4-22) is no longer an oracle formulation with respect to the optimization variables. Equivalently, Eq. (4-22) aims to minimizing the $\mathcal{L}_1$-norm of a vector containing the weighted modal property difference and the weighted generalized eigenvalue equation (Eq. (4-22b)) [36].

The optimization problem in Eq. (4-22) is still nonconvex with respect to the optimization variables, because of the matrix-vector multiplication in the constraint sets Eq. (4-22b). Nevertheless, the optimization problem in Eq. (4-22) is expressed explicitly with respect to the optimization variables, and the only nonconvex term in the optimization problem is the matrix-vector multiplication in Eq. (4-22b). In addition, it can be discovered

that only bilinear terms exist in Eq. (4-22b), so the convex-relaxation introduced in Eq. (4-3) is applicable. Thus, the reformulated eigenvector difference formulation can be solved by B&B algorithm. On the other hand, when the vector optimization variable $\boldsymbol{\delta}$, stiffness updating variables ($\boldsymbol{\alpha}$) as well as the simulated eigenvalues $\boldsymbol{\Lambda}$ are grouped as $\mathbf{x}$, and $\boldsymbol{\varphi}_{-q}^{m}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$ as $\mathbf{y}$, Eq. (4-22) has the form of a biconvex problem defined in Eq. (4-8). Therefore, the optimization problem can also be solved by the P-RD algorithm.

4.3.2 Modal Dynamic Residual Formulation

To apply the global optimization algorithms, the modal dynamic residual formulation defined in Eq. (2-10) is also reformulated. The reformulation adopts the updating variables $\boldsymbol{\alpha} \in \mathbb{R}^{n_{\boldsymbol{\alpha}}}$, the unmeasured DOFs in eigenvector $\boldsymbol{\varphi}^{\mathrm{u}} = [\boldsymbol{\psi}_1^{\mathrm{u}} \quad \boldsymbol{\psi}_2^{\mathrm{u}} \quad \cdots \quad \boldsymbol{\psi}_{n_{\mathrm{modes}}}^{\mathrm{u}}]^{\mathrm{T}} \in \mathbb{R}^{n_{\mathrm{u}} \cdot n_{\mathrm{modes}}}$, as well as a vector variable $\boldsymbol{\delta} \in \mathbb{R}^{N \cdot n_{\mathrm{modes}}}$ as optimization variables. The vector variable $\boldsymbol{\delta}$ equals $[\boldsymbol{\delta}_1 \quad \boldsymbol{\delta}_2 \quad \cdots \quad \boldsymbol{\delta}_{n_{\mathrm{modes}}}]^{\mathrm{T}}$, where $\boldsymbol{\delta}_i \in \mathbb{R}^N$.

$$\underset{\boldsymbol{\alpha},\boldsymbol{\varphi}^{\mathrm{u}},\boldsymbol{\delta}}{\text{minimize}} \quad \|\boldsymbol{\delta}\|_1 \tag{4-23a}$$

$$\text{subject to } -\boldsymbol{\delta}_i \leq \left[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}}\mathbf{M}\right]\begin{Bmatrix}\boldsymbol{\psi}_i^{\mathrm{EXP,m}} \\ \boldsymbol{\psi}_i^{\mathrm{u}}\end{Bmatrix} \cdot w_i \leq \boldsymbol{\delta}_i, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-23b}$$

$$\boldsymbol{L}_{\boldsymbol{\psi}_i^{\mathrm{u}}} \leq \boldsymbol{\psi}_i^{\mathrm{u}} \leq \boldsymbol{U}_{\boldsymbol{\psi}_i^{\mathrm{u}}}, \; i = 1 \ldots n_{\mathrm{modes}} \tag{4-23c}$$

$$\boldsymbol{L}_{\boldsymbol{\alpha}} \leq \boldsymbol{\alpha} \leq \boldsymbol{U}_{\boldsymbol{\alpha}} \tag{4-23d}$$

where the sign "≤" in Eq. (4-23b) represents element-wise inequality. Similar to the eigenvector difference formulation, Eq. (4-23) aims to minimize the $\mathcal{L}_1$-norm of the modal

dynamic residual vector, $\left[\mathbf{K}(\boldsymbol{\alpha}) - \lambda_i^{\mathrm{EXP}}\mathbf{M}\right]\begin{Bmatrix}\boldsymbol{\psi}_i^{\mathrm{EXP,m}}\\ \boldsymbol{\psi}_i^{\mathrm{u}}\end{Bmatrix}\cdot w_i, i = 1 \ldots n_{\mathrm{modes}}$. After reformulating, the modal dynamic residual formulation is still nonconvex, and the nonconvex term is the constraint set in Eq. (4-23b). Similar to the reformulated eigenvector difference formulation, the nonconvex parts in Eq. (4-23b) only have bilinear terms and thus can be applied with the convex-relaxation introduced in Eq. (4-3). The reformulated modal dynamic residual formulation can be solved by B&B algorithm. Furthermore, if the vector variable $\boldsymbol{\delta}$ and stiffness updating variables ($\boldsymbol{\alpha}$) are set as $\mathbf{x}$, and the unmeasured entries of eigenvectors ($\boldsymbol{\varphi}^{\mathrm{u}}$) as $\mathbf{y}$, Eq. (4-23) has the form of a biconvex problem, and thus can be solved by P-RD algorithm.

4.4 Validation Example

In this subsection, the performance of the model updating formulations rewritten for the two global optimization algorithms (Eqs. (4-22) and (4-23)) will be evaluated with both numerical simulations and experimental data. The numerical simulation is conducted on a plane truss structure, and the experimental data on the four-story shear-frame structure from CHAPTER 3 will be reused.

4.4.1 Plane Truss Structure

Figure 4-10 shows a plane truss structure, where all member sections are set as 8×10^{-5} m^2, and material density is set as 7,849 kg/m^3. The truss model has 10 nodes, and each node has a vertical and a horizontal DOF. Flexible support conditions are considered in this structure. Vertical and horizontal springs (k_{x1} and k_{y1}) are allocated at the left support, and a vertical spring (k_{y2}) is allocated at the right support. The Young's moduli of

the truss bars are divided into three group, i.e. E_1 for the top-level truss bars, E_2 for the diagonal and vertical truss bars, and E_3 for the bottom-level truss bars. Figure 4-10 also shows the instrumentation locations on the structure, where 8 DOFs are assumed to be measured, i.e. the length of both $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}$ is $n_{\mathrm{m}} = 8$.

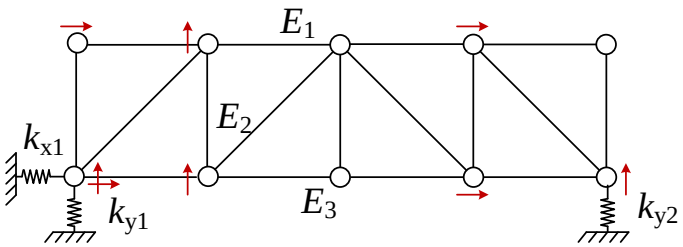


Figure 4-10. Plane truss structure and sensor instrumentation

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Table 4-1 lists the stiffness parameters to be updated, which are the stiffness values of top (E_1), diagonal & vertical (E_2) and bottom (E_3) truss members, as well as the support spring stiffness values (k_{y1}, k_{x1} and k_{y2}). The table also shows the nominal and true/actual stiffness parameter values. Corresponding to the 6 stiffness parameters being

Table 4-1 Structural properties of the plane-truss structure

Stiffness parameters		Nominal value	Actual value	Updating variables	α_i^{act}
Elastic moduli of truss members ($\times 10^{11}$ N/m^2)	E_1	2.0	2.2	α_1	0.100
	E_2	2.0	1.8	α_2	-0.100
	E_3	2.0	1.9	α_3	-0.050
Support spring stiffness ($\times 10^6$ N/m^2)	k_{y1}	6.0	7.0	α_4	0.167
	k_{x1}	6.0	3.0	α_5	-0.500
	k_{y2}	6.0	5.0	α_6	-0.167

updated, the $n_{\boldsymbol{\alpha}} = 6$ number of updating variables ($\alpha_1 \sim \alpha_6$) are listed in the table. Similar to CHAPTER 2 and CHAPTER 3, $\boldsymbol{\alpha}$ is scaled to represent the relative change percentage from the nominal value of each stiffness parameter. The column α_i^{act} in Table 4-1 lists the true/actual values of each α_i, i.e. the ideal solutions to be identified from FE model updating.

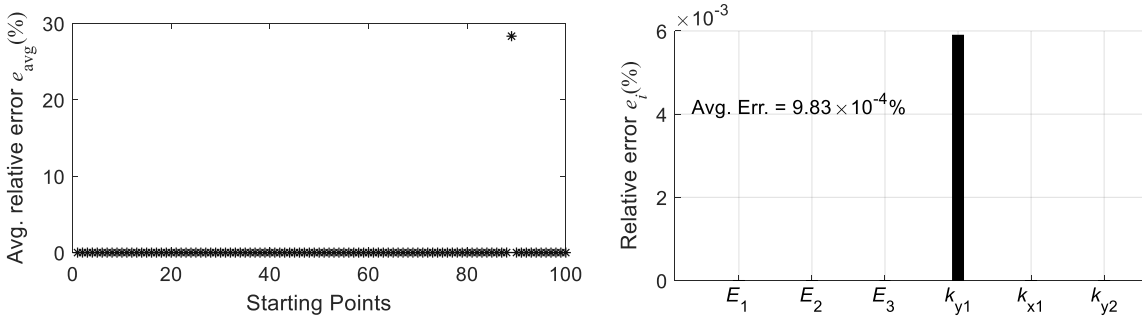
For this plane truss structure, modal properties of the structure with actual/correct stiffness values of $\boldsymbol{\alpha}$ are used as the "experimental" properties, i.e. λ_i^{EXP} and $\boldsymbol{\psi}_i^{\text{EXP,m}}$ in Eqs. (4-22) and (4-23). For practicality, only the mode associated with the lowest resonance frequency is assumed to be available for model updating, i.e. $n_{\text{modes}} = 1$. In this numerical study, the weighting factors in Eqs. (4-22b) and (4-23b) are all set as 1 for simplicity, i.e. $w_{\lambda_i} = w_{\boldsymbol{\psi}_i} = w_i = 1, i = 1 \ldots n_{\text{modes}}$. The weighting factor for the generalized eigenvalue equation in Eq. (4-22b) is set as 1×10^{-2}.

The model updating formulations in Eqs. (4-22) and (4-23) are first solved by the interior-point optimization algorithm implemented in the `fmincon` solver [76]. For both formulations, the lower and upper bounds of $\boldsymbol{\alpha}$ are set to be -1 and 1. Meanwhile, the lower and upper bounds of max-normalized $\boldsymbol{\varphi}_{-q}^{m}$ and $\boldsymbol{\varphi}^{\text{u}}$ are set to be -2 and 2. The lower and upper bounds for λ_i is set as $0.8\lambda_i^{\text{EXP}}$ and $1.2\lambda_i^{\text{EXP}}$. Since the global optimality of the solution from the interior-point optimization algorithm cannot be guaranteed, the optimization process is initiated from 100 random starting points of $\boldsymbol{\alpha}$, which are uniformly randomly generated between the upper and lower bounds. The initial value of $\boldsymbol{\varphi}^{\text{u}}$ in both eigenvector difference and modal dynamic residual formulations is obtained by solving the

generalized eigenvalue equation (Eq. (2-5)) with the random initial value of $\boldsymbol{\alpha}$. Finally, the initial values of λ_i and $\boldsymbol{\psi}_i^{\mathrm{m}}$ is always set equal to λ_i^{EXP} and $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$, respectively.

4.4.1.1 Updating Results of Eigenvector Difference Formulation

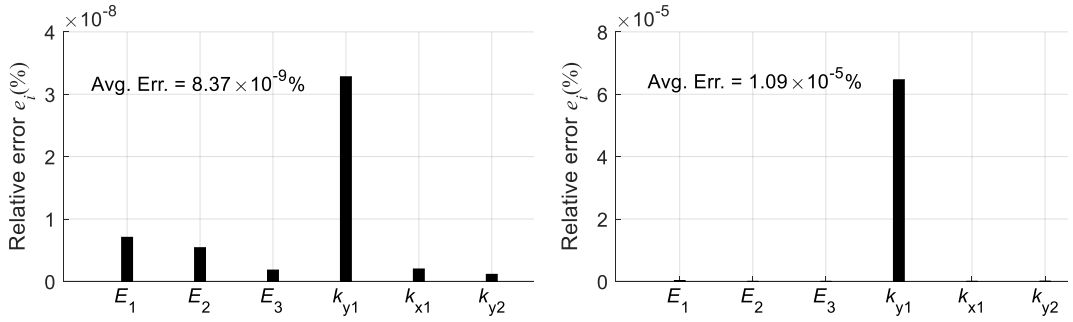
The optimization problem shown in Eq. (4-22) is first solved with the interior-point local optimization algorithm. Figure 4-11(a) shows the average relative error (e_{avg}) among all $n_{\boldsymbol{\alpha}} = 6$ optimal stiffness parameters, for each of the 100 optimal result sets. Horizontal axis is the sequence number of 100 randomized starting points. The figure shows that, 99 optimal result sets are close to the actual values of updating variables with e_{avg} smaller than 1%. Meanwhile, 1 optimization search ends up with unreasonable optimal result sets, i.e. $e_{\mathrm{avg}} > 28\%$. The inaccurate optimal result sets imply the nonconvexity of the eigenvector difference formulation. Additionally, the best solution is chosen as the set with the minimum objective function value. Figure 4-11(b) plots the relative error of each stiffness parameter (e_i) for the best solution. The average relative error in Figure 4-11(b) equals 9.83×10^{-3}%, which is negligible. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 60 seconds to obtain the 100 optimal result sets.



(a) Average of relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 4-11. Plane truss structure: updating results of eigenvector difference formulation (local optimization algorithm)

Alternatively, the optimization problem of the eigenvector difference formulation can be solved by two global optimization algorithms, i.e. B&B and P-RD, introduced before. Figure 4-12(a) and (b) respectively plot the relative error of each optimal stiffness parameter value obtained from B&B and P-RD algorithm. Both figures demonstrate that the optimal result set is almost identical to the actual value. Additionally, on this plane truss structure, the results from the global optimization algorithms are slightly more accurate than those from the local optimization algorithms. However, the difference is negligible from engineering point of view. When setting the tolerance for the difference between lower and upper bounds of optimal objective function value as 1×10^{-5}, it takes 0.20 seconds and 14 seconds for B&B and P-RD optimization algorithms to obtain the optimal result set.

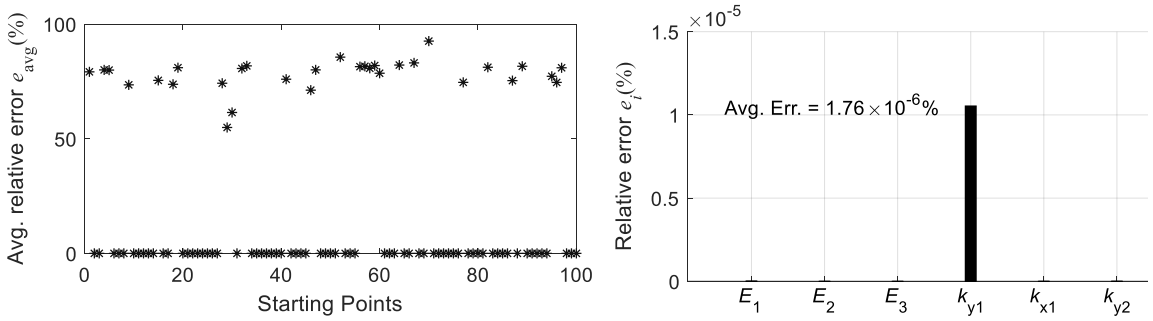


(a) Results of B&B algorithm (b) Results of P-RD algorithm

Figure 4-12. Plane truss structure: updating results of eigenvector difference formulation (global optimization algorithms)

4.4.1.2 Updating Results of Modal Dynamic Residual Formulation

The local optimization algorithm is then applied on the modal dynamic residual formulation in Eq. (4-23). Figure 4-13(a) plots e_{avg} of optimal stiffness parameter values for 100 optimal result sets. The figure shows that 69 optimal result sets can provide optimal



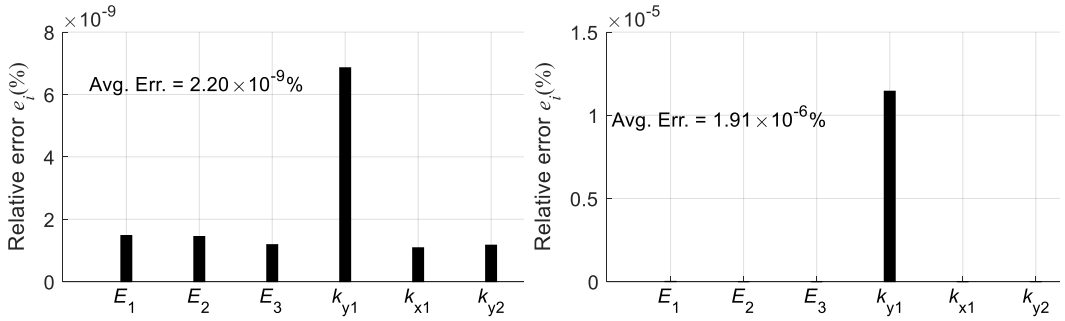
(a) Average of relative error of stiffness parameters (b) Relative error of the stiffness parameters

Figure 4-13. Plane truss structure: updating results of modal dynamic residual formulation (local optimization algorithm)

stiffness parameter values close to actual ones with $e_{\text{avg}} < 1\%$. The results for the other optimal solution sets are not reasonable, with $e_{\text{avg}} > 50\%$. Among 100 result sets, the best solution is selected as the one with the smallest objective function value. Figure 4-13(b) plots the relative error of each stiffness parameter for the best solution. The figure demonstrates that the best solution set of the modal dynamic residual formulation is better than the eigenvector difference formulation, with e_{avg} equal to 1.76×10^{-6}%. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it only takes 9 seconds to obtain the 100 optimal result sets.

The model updating of the plane truss is repeated using modal dynamic residual formulation, but with the B&B and P-RD global optimization algorithms. The relative error of each optimal stiffness parameter value from two global optimization algorithms is plotted in Figure 4-14. Again, the optimal values from both algorithms are demonstrated to be almost identical to actual ones. Compared to the best solution set from local optimization algorithms, the accuracy of result sets from B&B optimization algorithms are more accurate, but the difference is negligible from engineering point of view. When

setting the tolerance for the difference between lower and upper bounds of optimal objective function value as 1×10^{-5}, it takes 0.11 seconds and 0.63 seconds for B&B and P-RD optimization algorithms to obtain the optimal result set.



(a) Results of B&B algorithm (b) Results of P-RD algorithm

Figure 4-14. Plane truss structure: updating results of modal dynamic residual formulation (global optimization algorithms)

On this numerical simulation structure, the relative error of optimal result sets from 100 starting points shown in Figure 4-11(a) and Figure 4-13(a) demonstrate the nonconvexity of both reformulated eigenvector difference formulation (Eq. (4-22)) and the modal dynamic residual formulation (Eq. (4-23)). Thus, the global optimality of optimal solution from local optimization algorithm cannot be guaranteed. The optimal stiffness parameter values from inaccurate result sets can have e_{avg} larger than 40%. On the other hand, for both eigenvector difference and modal dynamic residual formulations, B&B and P-RD global optimization algorithms successfully converge around the actual values with negligible relative error.

4.4.2 Four-Story Shear-Frame Structure

In this subsection, the experimental data on the four-story shear-frame structure in CHAPTER 3 is reused to evaluate the performance of eigenvector difference and modal dynamic residual formulations with the B&B and P-RD global optimization algorithms.

To increase challenge of the model updating problem, in this subsection, the top floor of the shear-frame structure is assumed to be unmeasured, and thus only 3 DOFs of the structure are instrumented with sensor, i.e. the length of both $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$ and $\boldsymbol{\psi}_i^{\mathrm{m}}$ is $n_{\mathrm{m}} = 3$. Also, it is assumed that the modal properties of first two modes are available for model updating, i.e. $n_{\mathrm{modes}} = 2$. In summary, the available modal properties are plotted in Figure 4-15.

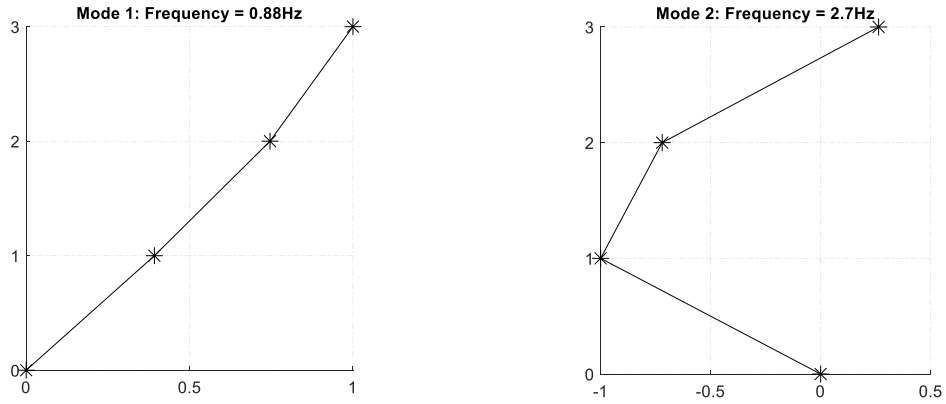


Figure 4-15. Modal properties of first two modes of the shear-frame structure

Same as CHAPTER 3, the FE model of the shear-frame frame structure is simplified as a lumped mass-spring structure, where all the spring stiffness values are set to 10 lbf/in and the mass of each DOF is set as 12.060 lb. The comparison of experimental and simulated modal properties of the nominal model is summarized in Table 4-2.

Table 4-2 Comparison of experimental and simulated modal properties from the nominal model

Modes	Experimental results	Nominal model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.99	12.02	1.00
2nd mode	2.75	2.85	3.64	0.93

It is assumed that the mass matrix (**M**) is accurate enough and does not require updating. Table 4-3 lists the stiffness parameters to be updated, which are the stiffness values of spring $k_1 \sim k_4$. Corresponding to the 4 stiffness parameters being updated, the $n_{\boldsymbol{\alpha}} = 4$ number of updating variables ($\alpha_1 \sim \alpha_4$) are also listed in the table. Both eigenvector difference formulation (Eq. (4-22)) and modal dynamic residual formulation (Eq. (4-23)) are adopted to update the FE model of this four-story shear-frame structure. For this laboratory structure, the weighting factors in Eqs. (4-22b) and (4-23b) are all set as 1 for simplicity, i.e. $w_{\lambda_i} = w_{\boldsymbol{\psi}_i} = w_i = 1, i = 1 \ldots n_{\mathrm{modes}}$. The weighting factor for the generalized eigenvalue equation in Eq. (4-22b) is set as 1×10^{-3}.

Table 4-3 Structural properties of the shear-frame structure

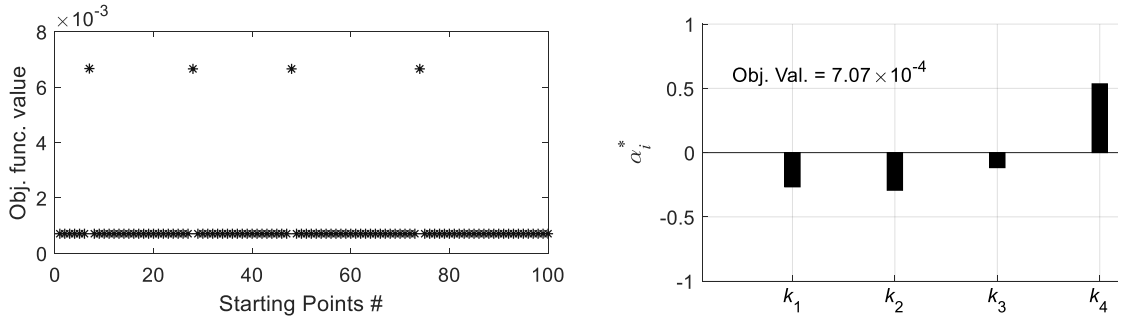
Stiffness parameters		Nominal value	Updating variables
Spring stiffness value (lbf/in)	k_1	10.00	α_1
	k_2	10.00	α_2
	k_3	10.00	α_3
	k_4	10.00	α_4

Similar to the plane truss structure, the model updating formulations are first solved by the interior-point optimization algorithm implemented in the `fmincon` solver. For both

formulations, the lower and upper bounds of $\boldsymbol{\alpha}$ are set to be -1 and 1. Meanwhile, the lower and upper bounds of $\boldsymbol{\varphi}_{-q}^{m}$ and $\boldsymbol{\varphi}^{\mathrm{u}}$ are set to be -2 and 2. The lower and upper bounds for λ_i is set as $0.8\lambda_i^{\mathrm{EXP}}$ and $1.2\lambda_i^{\mathrm{EXP}}$. Also, the optimization process is initiated from 100 random starting points of $\boldsymbol{\alpha}$, which are uniformly randomly generated between the upper and lower bounds. The initial value of $\boldsymbol{\varphi}^{\mathrm{u}}$ in both eigenvector difference and modal dynamic residual formulations is obtained by solving the generalized eigenvalue equation (Eq. (2-5)) with the random initial value of $\boldsymbol{\alpha}$. Finally, the initial values of λ_i and $\boldsymbol{\psi}_i^{\mathrm{m}}$ is always set equal to λ_i^{EXP} and $\boldsymbol{\psi}_i^{\mathrm{EXP,m}}$, respectively.

4.4.2.1 Updating Results of Eigenvector Difference Formulation

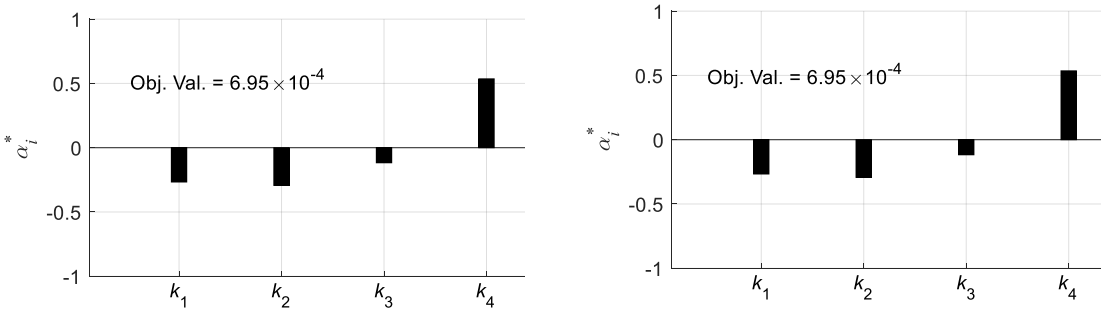
Figure 4-16(a) plots the optimal objective function values of 100 random starting points using the interior-point optimization algorithm. The figure demonstrates that most optimization processes can converge at the same optimal point. For the result set with the minimum objective function value among 100 optimal result sets, Figure 4-16(b) plots the optimal value of each updating parameter (α_i^*). The figure shows that among all the updating variables, only the inter-story stiffness value of top story increases, while the stiffness values of the other stories decrease, which is because of the P-Δ effect from the weight of aluminum plates. The overall trend for the change of inter-story stiffness is consistent with the results in CHAPTER 3. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 58 seconds to obtain the 100 optimal result sets.



(a) Objective function values of 100 optimal result sets (b) Optimal values of updating variables

Figure 4-16. Shear frame structure: updating results of eigenvector difference formulation (local optimization algorithm)

Alternatively, the optimization problem of the eigenvector difference formulation can be solved by B&B and P-RD global optimization algorithms. Figure 4-17(a) and (b) respectively plot the optimal updating variable values obtained from B&B and P-RD algorithms as well as the corresponding optimal objective function values. The figures demonstrate that the optimal result sets from two global optimization algorithms are identical. Additionally, the optimal objective function value is slightly smaller than the best solution set obtained from the local optimization algorithm in Figure 4-16(b), but the difference is negligible. When setting the tolerance for the difference between lower and



(a) Results of B&B algorithm (b) Results of P-RD algorithm

Figure 4-17. Shear frame structure: updating results of eigenvector difference formulation (global optimization algorithms)

upper bounds of optimal objective function value as 1×10^{-5}, it takes 0.17 seconds and 46 minutes for B&B and P-RD optimization algorithms to obtain the optimal result set.

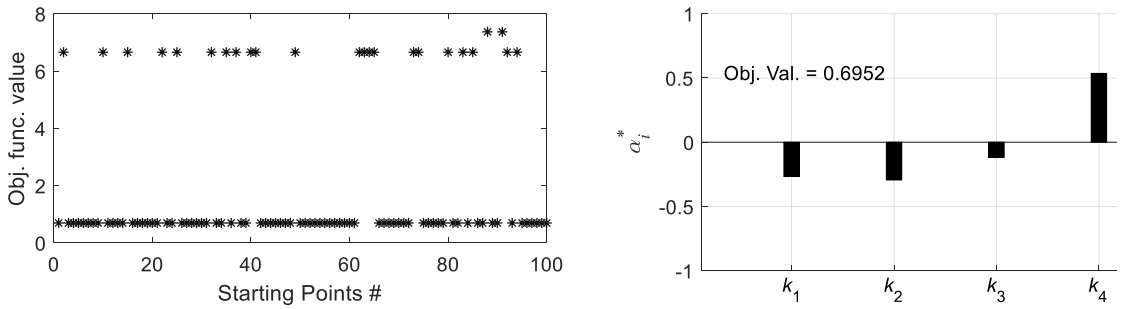
With α_i^* displayed in Figure 4-17, an updated FE model can be built. Table 4-4 compares the modal properties generated by the updated FE model with the experimental modal properties. Compared to Table 4-3, the frequency difference (Δf_i) and the MAC values of the eigenvectors at measured DOFs for the 1st and 2nd mode imply that the updated model is almost identical to the actual structure.

Table 4-4 Comparison of experimental and simulated modal properties from the eigenvector difference formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.88	1.91	1.00
2nd mode	2.75	2.75	0.00	1.00

4.4.2.2 Updating Results of Modal Dynamic Residual Formulation

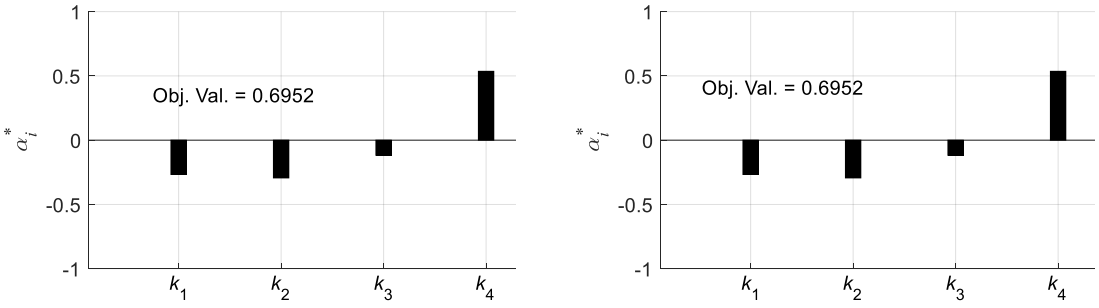
The local optimization algorithm is applied on the modal dynamic residual formulation, and Figure 4-18(a) plots the optimal objective function values of 100 optimal result sets. The figure shows that the optimal results from 24 starting points are unreasonable. For this best solution among the 100 result sets, the optimal value of $\boldsymbol{\alpha}$ is plotted in Figure 4-18(b). The figure demonstrates that similar to the results from the eigenvector difference formulation, the inter-story stiffness value of the top story is the highest. On a PC with an Intel i7-7700 CPU and 16 GB RAM, it takes 5 seconds to obtain the 100 optimal result.



(a) Objective function values of 100 optimal result sets (b) Optimal values of updating variables

Figure 4-18. Shear frame structure: updating results of modal dynamic residual formulation (local optimization algorithm)

The application of the modal dynamic residual formulation on the model updating of the shear-frame structure is repeated with the B&B and P-RD global optimization algorithms. The mode updating result sets from the B&B and P-RD algorithms are plotted in Figure 4-19(a) and (b), respectively. As expected, not only the optimal result sets from two global optimization algorithms are identical, but also the optimal result set is same as the best solution set using the local optimization algorithm shown in Figure 4-18(b). When setting the tolerance for the difference between lower and upper bounds of optimal objective function value as 1×10^{-5}, it takes 0.28 seconds and 0.16 seconds for B&B and P-RD optimization algorithms to obtain the optimal result set.



(a) Results of B&B algorithm (b) Results of P-RD algorithm

Figure 4-19. Shear frame structure: updating results of modal dynamic residual formulation (global optimization algorithms)

Similar to the eigenvector difference formulation, an updated FE model is built with the optimal updating variable values whose modal properties are summarized in Table 4-5. The table shows the low error in resonance frequency and high MAC value for both 1st and 2nd modes. This demonstrates that the updated model from the modal dynamic residual formulation is much closer to the actual structure than the nominal model.

Table 4-5 Comparison of experimental and simulated modal properties from the modal dynamic residual formulation

Modes	Experimental results	Updated model		
	f_i^{EXP} (Hz)	f_i (Hz)	Δf_i (%)	MAC
1st mode	0.88	0.87	1.91	1.00
2nd mode	2.75	2.75	0.00	1.00

On this laboratory shear-frame structure, the 100 optimal objective function values shown in Figure 4-16(a) and Figure 4-18(a) further demonstrate the nonconvexity of both the eigenvector difference formulation (Eq. (4-22)) and the modal dynamic residual formulation (Eq. (4-23)). As a result, for both formulations, optimization starting from some starting points cannot provide reasonable optimal stiffness parameter values. On the other hand, for both model updating formulations, the optimal results from B&B and P-RD global optimization algorithms agree with the best solution set from the local optimization algorithm. The agreement can confirm that the best solution sets from local optimization algorithm are global minimum within the feasible region of the corresponding optimization problem.

CHAPTER 5. SUMMARY AND FUTURE WORK

To reduce the discrepancy between the behaviour of a finite element (FE) model and the actual structure, a variety of FE model updating approaches can be adopted to update selected parameters of the FE model. This dissertation focuses on the FE model updating approaches based on frequency-domain modal properties, i.e. resonance frequencies and mode shapes. Formulated as optimization problems, the frequency-domain model updating approaches can be attempted with both local and global optimization algorithms. This chapter first provides a conclusion of the dissertation, and then discusses future research directions.

5.1 Summary and Conclusions

This research studies using both local and global optimizations to solve the frequency-domain model updating problems. The work and corresponding conclusions are summarized as follows:

1. To update selected stiffness parameters of an FE model, three frequency-domain FE model updating formulations, i.e. MAC value, eigenvector difference and modal dynamic residual formulations, are adopted. At first, two local optimization algorithms, i.e. Levenberg-Marquardt and trust-region-reflective algorithms, are utilized to solve the optimization problems of all three model updating formulations. To facilitate the optimization process, the analytical gradient of three model updating formulations is derived, including a new approach of obtaining the eigenvector gradient with respect to the updating variables. The numerical simulations demonstrate that the MAC value

formulation can correctly identify stiffness parameter values for a relatively simpler structural model (e.g. the lumped mass-spring and steel pedestrian bridge), but fails to provide reasonable results when the model complexity increases (e.g. the concrete building frame). On the other hand, the eigenvector difference and modal dynamic residual formulations are able to correctly update the structural parameter values for all three structural models. Comparing two local optimization algorithms, the Levenberg-Marquardt algorithm implemented in MATLAB `lsqnonlin` performs more efficiently and can be applied when the optimization problem is underdetermined. However, the implementation does not allow setting bounds of updating variables. On the other hand, the trust-region-reflective algorithm implemented in MATLAB `lsqnonlin` cannot be applied to underdetermined problems, but can ensure the optimization search results are within bounds. In addition, using analytical gradient during the optimization process in general not only provides more accurate model updating results, but also saves computing time.

2. In order to collect vibration data from actual structures at low cost, a wireless sensing node, *Martlet*, and an integrated accelerometer board are developed. The performance of wireless sensing system is demonstrated to be reliable through both laboratory and field experiments. Experimental modal properties of the actual structure are successfully extracted from the collected structural vibration data. For the laboratory shear-frame structure, MAC value, eigenvector difference and modal dynamic residual formulations provide similar updated inter-story stiffness values. For the steel pedestrian bridge, the statistical weightings of the identified modal properties are incorporated into the MAC value and eigenvector difference formulations. Again, the updated models provided by the MAC value and eigenvector difference formulation are considered to be

close to each other. For both structures (the laboratory shear-frame structure and the steel pedestrian bridge), the simulated modal properties of the updated FE model are much closer to the experimental modal properties than those from the nominal FE model.

3. The presented model updating formulations are in general nonconvex with unknown number of local minima, and the numerical simulations further confirm the nonconvexity of the model updating formulations. The global optimality of the optimal solution from local optimization algorithms cannot be guaranteed. As a result, this research also explores solving the FE model updating problems with two global optimization algorithms, i.e. branch-and-bound (B&B) and primal-relaxed dual (P-RD) algorithms. To apply the global optimization algorithms to the FE model updating problems, the eigenvector difference and modal dynamic residual formulations are reformulated. Validated with both numerical and experimental data, both B&B and P-RD algorithms are able to correctly obtain the global minimum of the reformulated model updating formulations. Comparing two global optimization algorithms, B&B algorithm is demonstrated to be more efficient than P-RD algorithm on the presented examples.

5.2 Future Work

Based on the current achievements, future research can be expanded along following directions.

1. The FE model updating formulations studied in this research can only update the stiffness and mass parameter of the structural model. Under certain circumstance, the damping information of the structure is also of great importance. Therefore, investigation is needed to update the damping information of the structural model, such as finding

appropriate damping parameters for the structural model and developing model updating formulations involving the structural damping matrix.

2. The presented frequency-domain model updating formulations have been validated on an FE model with up to 2,302 degrees of freedom (DOF), i.e. the numerical simulation on the concrete building frame. Studies can be conducted on applying the presented model updating formulations on more complicated structural models with a larger number of DOFs. Inevitably, the computational time will become a big challenge when the complexity of the structural model increases. Accordingly, more emphasis can be placed on increasing the computational efficiency, such as finding more effective local optimization algorithms to reduce the number of iterations. On the other hand, the efficiency of applying nonsmooth optimization algorithms on the FE model updating problems, such as bundle method [81], can be investigated. The bundle method constructs a piecewise linear approximation to the objective function, and the search direction is determined by the approximation.

3. This research demonstrates that when applied with field experimental data, the MAC value and eigenvector difference formulations are able to provide updated FE models with behaviors closer to the actual structure. If reference baseline data is available, FE model updating may be used to identify parameter value changes caused by structural deterioration or damage. Future studies can also be conducted to automate the FE model updating using the sensor data collected from a structure in real time.

4. The B&B and P-RD global optimization algorithms with the reformulated model updating formulations should be validated through more complicated structural models.

Especially for the P-RD algorithm, future study is needed to increase the efficiency of the optimization process. For example, at each iteration, the P-RD algorithm requires solving relaxed dual subproblems for all the possible combinations of lower and upper bounds for **x** variables. Solving those relaxed dual subproblems does not need to be sequential, and thus parallel computing can be adopted. In addition, the application of the global optimization algorithms on other frequency-domain model updating formulations, such as modal flexibility and curvature mode shapes, can also be interesting future directions.

REFERENCES

[1] J. P. Lynch and K. J. Loh, "A summary review of wireless sensors and sensor networks for structural health monitoring," *The Shock and Vibration Digest*, vol. 38, pp. 91-128, 2006.

[2] M. Kane, D. Zhu, M. Hirose, X. Dong, B. Winter, M. Häckell, J. P. Lynch, Y. Wang, and A. Swartz, "Development of an extensible dual-core wireless sensing node for cyber-physical systems," in *Proceedings of SPIE, Nondestructive Characterization for Composite Materials, Aerospace Engineering, Civil Infrastructure, and Homeland Security*, p. 90611U, San Diego, California, USA, 2014.

[3] R. A. Swartz, D. Jung, J. P. Lynch, Y. Wang, D. Shi, and M. P. Flynn, "Design of a wireless sensor for scalable distributed in-network computation in a structural health monitoring system," in *Proceedings of the 5th International Workshop on Structural Health Monitoring*, Stanford, CA, 2005.

[4] M. I. Friswell and J. E. Mottershead, *Finite Element Model Updating in Structural Dynamics*, Dordrecht; Boston: Kluwer Academic Publishers, 1995.

[5] M. Hoshiya and E. Saito, "Structural identification by extended Kalman filter," *Journal of Engineering Mechanics-ASCE*, vol. 110, pp. 1757-1770, 1984.

[6] J. N. Yang, S. Pan, and H. Huang, "An adaptive extended Kalman filter for structural damage identifications II: unknown inputs," *Structural Control and Health Monitoring*, vol. 14, pp. 497-521, 2007.

[7] L. Zhou, S. Wu, and J. N. Yang, "Experimental study of an adaptive extended Kalman filter for structural damage identification," *Journal of Infrastructure Systems*, vol. 14, pp. 42-51, 2008.

[8] H. Ebrahimian, R. Astroza, and J. P. Conte, "Extended Kalman filter for material parameter estimation in nonlinear structural finite element models using direct differentiation method," *Earthquake Engineering & Structural Dynamics*, vol. 44, pp. 1495-1522, 2015.

[9] A. W. Smyth, S. F. Masri, A. G. Chassiakos, and T. K. Caughey, "On-line parametric identification of MDOF nonlinear hysteretic systems," *Journal of Engineering Mechanics,* vol. 125, pp. 133-142, 1999.

[10] A. W. Smyth, S. F. Masri, E. B. Kosmatopoulos, A. G. Chassiakos, and T. K. Caughey, "Development of adaptive modeling techniques for non-linear hysteretic systems," *International Journal of Non-Linear Mechanics,* vol. 37, pp. 1435-1451, 2002.

[11] Q. W. Zhang, C. C. Chang, and T. Y. P. Chang, "Finite element model updating for structures with parametric constraints," *Earthquake Engineering & Structural Dynamics,* vol. 29, pp. 927-944, 2000.

[12] O. S. Salawu, "Detection of structural damage through changes in frequency: A review," *Engineering Structures,* vol. 19, pp. 718-723, 1997.

[13] P. W. Moller and O. Friberg, "Updating large finite element models in structural dynamics," *AIAA Journal,* vol. 36, pp. 1861-1868, 1998.

[14] D. Ribeiro, R. Calçada, R. Delgado, M. Brehm, and V. Zabel, "Finite element model updating of a bowstring-arch railway bridge based on experimental modal parameters," *Engineering Structures,* vol. 40, pp. 413-435, 2012.

[15] A. Teughels, J. Maeck, and G. De Roeck, "Damage assessment by FE model updating using damage functions," *Computers & Structures,* vol. 80, pp. 1869-1879, 2002.

[16] S.-E. Fang, R. Perera, and G. De Roeck, "Damage identification of a reinforced concrete frame by finite element model updating using damage parameterization," *Journal of Sound and Vibration,* vol. 313, pp. 544-559, 2008.

[17] B. Jaishi and W. X. Ren, "Damage detection by finite element model updating using modal flexibility residual," *Journal of Sound and Vibration,* vol. 290, pp. 369-387, 2006.

[18] S. Zhou and W. Song, "Environmental‐effects‐embedded model updating method considering environmental impacts," *Structural Control and Health Monitoring,* vol. 25, p. e2116, 2018.

[19] K. V. Yuen, "Updating large models for mechanical systems using incomplete modal measurement," *Mechanical Systems and Signal Processing,* vol. 28, pp. 297-308, 2012.

[20] C. Farhat and F. M. Hemez, "Updating finite element dynamic models using an element-by-element sensitivity methodology," *AIAA Journal,* vol. 31, pp. 1702-1711, 1993.

[21] M. Sanayei, J. A. McClain, S. Wadia-Fascetti, and E. M. Santini, "Parameter estimation incorporating modal data and boundary conditions," *Journal of structural engineering,* vol. 125, pp. 1048-1055, 1999.

[22] M. Sanayei, B. Arya, E. M. Santini, and S. Wadia‐Fascetti, "Significance of modeling error in structural parameter estimation," *Computer‐Aided Civil and Infrastructure Engineering,* vol. 16, pp. 12-27, 2001.

[23] M. O. Abdalla, K. M. Grigoriadis, and D. C. Zimmerman, "Structural damage detection using linear matrix inequality methods," *Journal of Vibration and Acoustics,* vol. 122, pp. 448-455, 2000.

[24] D. Zhu, X. Dong, and Y. Wang, "Substructure stiffness and mass updating through minimization of modal dynamic residuals," *Journal of Engineering Mechanics,* vol. 142, p. 04016013, 2016.

[25] J. D. Sipple and M. Sanayei, "Finite element model updating using frequency response functions and numerical sensitivities," *Structural Control and Health Monitoring,* vol. 21, pp. 784-802, 2014.

[26] R. Lin and D. Ewins, "Analytical model improvement using frequency response functions," *Mechanical Systems and Signal Processing,* vol. 8, pp. 437-458, 1994.

[27] H. Lus, R. Betti, J. Yu, and M. De Angelis, "Investigation of a system identification methodology in the context of the ASCE benchmark problem," *Journal of Engineering Mechanics,* vol. 130, pp. 71-84, 2004.

[28] H. Ahmadian, J. E. Mottershead, and M. I. Friswell, "Regularisation methods for finite element model updating," *Mechanical Systems and Signal Processing,* vol. 12, pp. 47-64, 1998.

[29] J. L. Beck and L. S. Katafygiotis, "Updating Models and Their Uncertainties. I: Bayesian Statistical Framework," *Journal of Engineering Mechanics,* vol. 124, pp. 455-461, 1998.

[30] K.-V. Yuen and S.-C. Kuok, "Bayesian methods for updating dynamic models," *Applied Mechanics Reviews,* vol. 64, p. 010802, 2011.

[31] L. S. Katafygiotis, C. Papadimitriou, and H.-F. Lam, "A probabilistic approach to structural model updating," *Soil Dynamics and Earthquake Engineering,* vol. 17, pp. 495-507, 1998.

[32] J. Ching, M. Muto, and J. L. Beck, "Structural model updating and health monitoring with incomplete modal data using Gibbs sampler," *Computer‐Aided Civil and Infrastructure Engineering,* vol. 21, pp. 242-257, 2006.

[33] I. Behmanesh and B. Moaveni, "Probabilistic identification of simulated damage on the Dowling Hall footbridge through Bayesian finite element model updating," *Structural Control and Health Monitoring,* vol. 22, pp. 463-483, 2015.

[34] J. E. Mottershead, M. Link, and M. I. Friswell, "The sensitivity method in finite element model updating: A tutorial," *Mechanical Systems and Signal Processing,* vol. 25, pp. 2275-2296, 2011.

[35] MathWorks Inc., *Optimization Toolbox™ User's Guide,* R2015b ed., Natick, MA, 2015.

[36] S. P. Boyd and L. Vandenberghe, *Convex Optimization*: Cambridge University Press, 2004.

[37] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics,* vol. 21, pp. 1087-1092, 1953.

[38] R. A. Fisher, *The Genetical Theory of Natural Selection: A Complete Variorum Edition*: Oxford University Press, 1999.

[39] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science,* pp. 39-43, 1995.

[40] T. Marwala, *Finite element model updating using computational intelligence techniques: applications to structural dynamics*: Springer Science & Business Media, 2010.

[41] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part I — Convex underestimating problems," *Mathematical Programming,* vol. 10, pp. 147-175.

[42] H. S. Ryoo and N. V. Sahinidis, "A branch-and-reduce approach to global optimization," *Journal of Global Optimization,* vol. 8, pp. 107-138, 1996.

[43] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical Programming,* vol. 103, pp. 225-249, 2005.

[44] S. S. Dey, A. Santana, and Y. Wang, "New SOCP relaxation and branching rule for bipartite bilinear programs," *Optimization and Engineering,* vol. in print, 2018.

[45] C. Floudas and V. Visweswaran, "A global optimization algorithm (GOP) for certain classes of nonconvex NLPs—I. Theory," *Computers & Chemical Engineering,* vol. 14, pp. 1397-1417, 1990.

[46] C. A. Floudas and V. Visweswaran, "Primal-relaxed dual global optimization approach," *Journal of Optimization Theory and Applications,* vol. 78, pp. 187-225, 1993.

[47] C. A. Floudas, *Deterministic Global Optimization: Theory, Methods and Applications* vol. 37: Springer Science & Business Media, 2013.

[48] A. Barvinok, *A Course in Convexity* vol. 54, Providence, RI: American Mathematical Society, 2002.

[49] M. Berger, "Convexity," *American Mathematical Monthly,* vol. 97, pp. 650-78, 1990.

[50] R. J. Allemang and D. L. Brown, "A correlation coefficient for modal vector analysis," in *Proceedings of the 1st International Modal Analysis Conference*, pp. 110-116, 1982.

[51] R. Fox and M. Kapoor, "Rates of change of eigenvalues and eigenvectors," *AIAA Journal*, vol. 6, pp. 2426-2429, 1968.

[52] R. B. Nelson, "Simplified calculation of eigenvector derivatives," *AIAA Journal*, vol. 14, pp. 1201-1205, 1976.

[53] M. I. Friswell, "Calculation of second and higher order eigenvector derivatives," *Journal of Guidance, Control, and Dynamics*, vol. 18, pp. 919-921, 1995.

[54] J. Moré, "The Levenberg-Marquardt algorithm: Implementation and Theory," in *Numerical Analysis*. vol. 630, G. A. Watson, Ed.: Springer Berlin Heidelberg, 1978, pp. 105-116.

[55] T. F. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, pp. 418-445, 1996.

[56] R. H. Byrd, R. B. Schnabel, and G. A. Shultz, "Approximate solution of the trust region problem by minimization over two-dimensional subspaces," *Mathematical Programming*, vol. 40, pp. 247-263, 1988.

[57] M. A. Branch, T. F. Coleman, and Y. Li, "A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems," *SIAM Journal on Scientific Computing*, vol. 21, pp. 1-23, 1999.

[58] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*: SIAM, 2007.

[59] X. Dong, X. Liu, T. Wright, Y. Wang, and R. DesRoches, "Validation of wireless sensing technology densely instrumented on a full-scale concrete frame structure," in *Proceedings of International Conference on Smart Infrastructure and Construction (ICSIC)*, Cambridge. United Kingdom, 2016.

[60] J. M. Ko and Y. Q. Ni, "Technology developments in structural health monitoring of large-scale bridges," *Engineering Structures*, vol. 27, pp. 1715-1725, 2005.

[61] C. R. Farrar, H. Sohn, F. M. Hemez, M. C. Anderson, M. T. Bement, P. J. Cornwell, S. W. Doebling, J. F. Schultze, N. Lieven, and A. N. Robertson, "Damage Prognosis:

Current Status and Future Needs," Los Alamos National Laboratory, Los Alamos, NM Report No. LA-14051-MS, 2003.

[62] J. N. Juang and R. S. Pappa, "An eigensystem realization algorithm for modal parameter identification and modal reduction," *Journal of Guidance Control and Dynamics,* vol. 8, pp. 620-627, 1985.

[63] B. Peeters and C. E. Ventura, "Comparative study of modal analysis techniques for bridge dynamic characteristics," *Mechanical Systems and Signal Processing,* vol. 17, pp. 965-988, 2003.

[64] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems: Theory—Implementation—Applications*: Springer Science & Business Media, 2012.

[65] M. Çelebi, "Seismic Instrumentation of Buildings (with Emphasis on Federal Buildings)," United States Geological Survey, Menlo Park, CA Report No. 0-7460-68170, 2002.

[66] R. A. Swartz, D. Jung, J. P. Lynch, Y. Wang, D. Shi, and M. P. Flynn, "Design of a wireless sensor for scalable distributed in-network computation in a structural health monitoring system," in *Proceedings of the 5th International Workshop on Structural Health Monitoring*, Stanford, CA, 2005.

[67] B. F. Spencer, H. Jo, K. A. Mechitov, J. Li, S.-H. Sim, R. E. Kim, S. Cho, L. E. Linderman, P. Moinzadeh, and R. K. Giles, "Recent advances in wireless smart sensors for multi-scale monitoring and control of civil infrastructure," *Journal of Civil Structural Health Monitoring*, vol. 6, pp. 17-41, 2016.

[68] Y. Wang, J. P. Lynch, and K. H. Law, "A wireless structural health monitoring system with multithreaded sensing devices: design and validation," *Structure and Infrastructure Engineering,* vol. 3, pp. 103-120, 2007.

[69] T. Nagayama and B. F. Spencer, Jr., "Structural Health Monitoring using Smart Sensors," Newmark Structural Engineering Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL Report No. NSEL-001, 2007.

[70] X. Dong, D. Zhu, Y. Wang, J. P. Lynch, and R. A. Swartz, "Design and validation of acceleration measurement using the Martlet wireless sensing system," in *Proceedings of the ASME 2014 Conference on Smart Materials, Adaptive Structures and Intelligent Systems (SMASIS)*, Newport, RI, USA, 2014.

[71] M. Kane, D. Zhu, M. Hirose, X. Dong, B. Winter, M. Häckell, J. P. Lynch, Y. Wang, and A. Swartz, "Development of an extensible dual-core wireless sensing node for cyber-physical systems " in *Proceedings of SPIE, Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems*, San Diego, CA, USA, 2014.

[72] T. Cooklev, *Wireless Communication Standards : a Study of IEEE 802.11, 802.15, and 802.16*, New York: Standards Information Network IEEE Press, 2004.

[73] P. Horowitz and W. Hill, *The Art of Electronics*, 2nd ed., Cambridge, England: Cambridge University Press, 1989.

[74] J. N. Juang and R. S. Pappa, "An eigensystem realization algorithm for modal parameter identification and model reduction," *Journal of Guidance Control and Dynamics*, vol. 8, pp. 620-627, 1985.

[75] N. R. Draper and H. Smith, *Applied Regression Analysis* vol. 326: John Wiley & Sons, 2014.

[76] R. H. Byrd, M. E. Hribar, and J. Nocedal, "An interior point algorithm for large-scale nonlinear programming," *SIAM Journal on Optimization*, vol. 9, pp. 877-900, 1999.

[77] G. P. McCormick, *Nonlinear Programming: Theory, Algorithms, and Applications*: John Wiley & Sons, 1983.

[78] M. Tawarmalani and N. V. Sahinidis, "Global optimization of mixed-integer nonlinear programs: A theoretical and computational study," *Mathematical programming*, vol. 99, pp. 563-591, 2004.

[79] N. V. Sahinidis, "BARON: Global Optimization of Mixed-Integer Nonlinear Programs," 2017.

[80] A. M. Geoffrion, "Generalized benders decomposition," *Journal of Optimization Theory and Applications*, vol. 10, pp. 237-260, 1972.

[81] M. Mäkelä, "Survey of bundle methods for nonsmooth optimization," *Optimization Methods and Software*, vol. 17, pp. 1-29, 2002.